VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

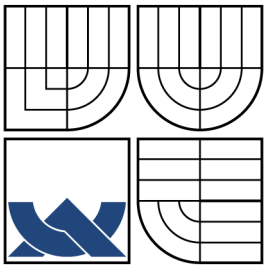
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF BIOMEDICAL ENGINEERING

POKROČILÉ DOLOVÁNÍ V DATECH V KARDIOLOGII

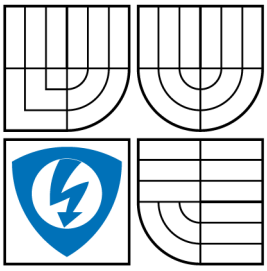
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE MARTIN MÉZL
AUTHOR

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF BIOMEDICAL ENGINEERING

POKROČILÉ DOLOVÁNÍ V DATECH V KARDIOLOGII
ADVANCED DATA MINING IN CARDIOLOGY

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — MARTIN MÉZL
AUTHOR

VEDOUCÍ PRÁCE — ING. JIŘÍ SEKORA
SUPERVISOR

BRNO 2009

ZDE VLOŽIT LIST ZADÁNÍ

Z důvodu správného číslování stránek

ZDE VLOŽIT PRVNÍ LIST LICENČNÍ SMOUVY

Z důvodu správného číslování stránek

ZDE VLOŽIT DRUHÝ LIST LICENČNÍ SMOUVY

Z důvodu správného číslování stránek

ABSTRAKT

Tato práce je zaměřena na využití data miningových metod v lékařství, konkrétně na databázi kardiologických pacientů. Cílem této práce je provést analýzu dat a zaměřit se na hledání neobvyklých závislostí mezi jednotlivými atributy souboru. Součástí práce je přehled dostupných metod, které se využívají v lékařství. Z těchto metod jsou pro další práci vybrány metody rozhodovacích stromů, naivního bayesovského klasifikátoru, umělých neuronových sítí a asociačních pravidel. Pro samotné hledání závislostí byly použity metody naivního bayesovského klasifikátoru a asociačních pravidel. Výstupem této práce je komplexní systém pro dobývání znalostí z databází na libovolném datovém souboru. Práce vznikla ve spolupráci s Interní kardiologickou klinikou Fakultní nemocnice Brno Bohunice. Všechny popsané aplikace byly vytvořeny v programovém prostředí Matlab 7.0.1.

KLÍČOVÁ SLOVA

dobývání znalostí z databází, data mining, dolování v datech, kardiologie, naivní bayesovský klasifikátor, rozhodovací stromy, asociační pravidla, umělé neuronové stromy

ABSTRACT

The aim of this master´s thesis is to analyse and search unusual dependencies in database of patients from Internal Cardiology Clinic Faculty Hospital Brno. The part of the work is theoretical overview of common data mining methods used in medicine, especially decision trees, naive Bayesian classifier, artificial neural networks and association rules. Looking for unusual dependencies between atributes is realized by association rules and naive Bayesian classifier. The output of this work is a complex system for Knowledge discovery in databases process for any data set. This work was realized with collaboration of Internal Cardiology Clinic Faculty Hospital Brno. All programs were made in Matlab 7.0.1.

KEYWORDS

Knowledge Discovery in Databases, Data Mining, Cardiology, Naive Bayesian Classifier, Decision Trees, Association Rules, Artificial Neural Networks

MÉZL M. *Pokročilé dolování v datech v kardiologii*. Brno: Vysoké učení technické. Fakulta elektrotechniky a komunikačních technologií. Ústav biomedicínského inženýrství, 2009. Počet stran 63, počet stran příloh 6. Diplomová práce. Vedoucí práce Ing. Jiří Sekora.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Pokročilé dolování v datech v kardiologii“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne ....

(podpis autora)

PODĚKOVÁNÍ

Chtěl bych tímto poděkovat za cenné připomínky, rady, materiály a vedení Ing. Jiřímu Sekorovi a externímu konzultantovi MUDr. Milanu Sepšimu, Ph.D.

OBSAH

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

ÚVOD

Dolování v datech (data mining) je v současné době jedním z nejmocnějších nástrojů pro analýzu dat. Od svého vzniku prošel velkým vývojem a postupně se vyčlenil z vědního oboru statistiky. Data mining nachází uplatnění v mnoha oblastech lidského života a můžeme říci, že se stal nedílnou součástí lidského života. Můžeme se s ním setkat při predikci vývoje kurzů akcií, při předpovědích počasí, v různých klasifikačních úlohách a také v medicíně. Využití v medicíně je rozmanité, od ekonomických úloh, kde může sloužit jako nástroj pro lepší financování, až po využití jako podpůrný prostředek pro diagnózu a terapii. Poznatky získané procesem dolování v datech se často implementují ve formě znalostí do expertních a informačních systémů.

Cílem této práce je provést analýzu datového souboru z Interní kardiologické kliniky Fakultní nemocnice Brno Bohunice. Tato analýza bude zaměřena na hledání závislostí mezi jednotlivými atributy dat. V úvodní části je rozebrána problematika procesu dobývání znalostí z databází. V další části jsou podrobně rozebrány jednotlivé dataminingové metody se zaměřením na využití v lékařství včetně problémů, které souvisí s charakterem dat. Další část je věnována realizaci konkrétních metod pro modelování — jsou to rozhodovací stromy, bayesovské metody, umělé neuronové sítě a asociační pravidla. Správnost realizovaných algoritmů byla ověřena pomocí typového příkladu. Poslední část je zaměřena na realizaci automatického hledání závislostí v datech pomocí metody naivního bayesovského klasifikátoru a asociačních pravidel.

1 DATA MINING

1.1 Získávání znalostí z databází

Historické počátky data miningu lze nalézt v 60. letech 20. století. Souvisely s využíváním prvních počítačů na vědecké půdě. Přesto byl vývoj data miningu brzděn dosahovaným výpočetním výkonem tehdejších počítačů. Získané postupy sloužily pouze pro výzkumné účely, zavádění postupů do praxe bylo velmi ojedinělé. V 90. letech minulého století nastal rozmach umělé inteligence (přesněji strojového učení). A právě metody strojového učení ve spojení s databázovými technologiemi umožnily další rozvoj data miningu. Právě v této době se v USA objevil pojem dobývání znalostí z databází (Knowledge Discovery in Databases, KDD). Důvodem vzniku tohoto pojmu bylo spojení dvou doposud samostatně se vyvíjejících odvětví a to databázových technologií a statistiky. Databázové technologie umožňují uchování velkého množství dat a hledání informací v nich. Toto uchování bylo v té době převratné a do té doby prakticky nemyslitelné. Statistika umožňuje analýzu dat a hledání souvislostí v datech. Původním cílem data miningu byla podpora pro strategické rozhodování ve firmách, v dnešní době se možnosti využití rozšířily do většiny sfér běžného života.[2]

V současné době chápeme pojem data mining jako samotný proces „dolování dat“ v rozsáhlých databázích, na rozdíl od pojmu dobývání znalostí z databází, který chápeme v poněkud širším významu. Dobývání znalostí z databází je interaktivní a iterativní proces tvořený kroky selekce, předzpracování, transformace, data mining a interpretace znalostí. Jiná definice říká, že dobývání znalostí z databází je možno chápat jako netriviální získávání implicitních, dříve neznámých a potenciálně užitečných informací z dat.[2, 9]

Nemůžeme říct, že by data mining měl jeden určitý cíl. Jedná se spíše o soubor metod a postupů, které slouží k analýze rozsáhlých databází. Základní dataminingovou úlohou je klasifikace, kdy se snažíme ze znalosti případů v určité skupině případů vyvodit obecná pravidla, která popisují chování této skupiny. Zpravidla se jedná o učení s učitelem, tzn. že posuzujeme závislosti mezi několika atributy a zvoleným cílovým atributem. Klasifikaci lze také použít jako úlohu učení bez učitele, kdy hledáme mezi atributy nějaký, předem neznámý, atribut, který dobře reprezentuje vztahy mezi ostatními atributy. Příkladem klasifikace je analýza klientů banky, kterým chceme poskytnout úvěr, nebo odhad diagnózy podle anamnézy a jednotlivých vyšetření u pacientů. Další typickou úlohou je predikce hodnot, kdy ze znalosti předchozích hodnot můžeme odhadnout hodnoty následující. Příkladem této úlohy je predikce kurzu akcií, popř. vývoj počasí. Cest k těmto cílům vede několik, záleží na charakteru dat, časových možnostech a také na odhadu analytika, který analýzu

provádí. Před uvedením do praxe je potřeba podrobit získané znalosti rozboru a testování. To nejčastěji provádíme tak, že celý datový soubor ještě před provedením analýzy rozdělíme na data trénovací a testovací. Na trénovacích datech provádíme samotnou analýzu a na testovacích datech ověřujeme získané znalosti. Zde se může projevit problém známý z využití umělých neuronových sítí v umělé inteligenci a to přeučení systému na konkrétní úlohu. To je skoro vždy nežádoucí, protože pomocí data miningu se snažíme najít obecné závislosti a vztahy mezi daty. Testování získaných znalostí je důležité především v lékařství, protože získané znalosti mohou pomáhat při rozhodování lékařů a jakákoli chyba je nepřípustná.[6, 14]

1.2 Metodiky data miningu

Pro řešení úloh v oblasti dobývání znalostí z databází vzniklo postupem času mnoho metodik, které poskytují obecný standard pro řešení úloh. Mezi nejznámější patří metodiky 5A (akronym od slov Assess, Access, Analyze, Act, Automate) od firmy SPSS a metodika SEMMA (akronym od slov Sample, Explore, Modify, Model, Assess) od firmy SAS. Tyto metodiky jsou produktem komerčních společností, které je používají jako základ svých programových systémů. Další metodikou, která bude blíže popsána, je metodika CRISP-DM, která vznikla ve spolupráci výzkumných a komerčních institucí. Podle této metodiky se budu snažit postupovat ve své další práci.[2, 6]

Metodika CRISP-DM (Cross-Industry Standard Process for Data Mining) vznikla v rámci Evropského výzkumného projektu. Cílem tohoto projektu bylo navrhnout univerzální postup pro dobývání znalostí z databází, který bude použitelný v komerčních aplikacích. Součásti tohoto projektu bylo také vytvoření „průvodce“ pro řešení problémů, které se mohou vyskytnout v reálných případech. Na vývoji této metodiky spolupracovaly firmy NCR (dodavatel datových skladů), DaimlerChrysler, ISL (tvůrce komerčního systému Clementine), OHRA (holandská pojišťovna) a další, které mají mnoho zkušeností s řešením reálných aplikací. Vývoj této metody začal na konci roku 1996 a byl dokončen v polovině roku 1999.

Celý proces dobývání znalostí z databází je podle metodiky CRISP-DM tvořen šesti fázemi viz obr. 1.2. Pořadí jednotlivých kroků není pevně dáno. Jedná se o iterativní postup, kdy se můžeme vracet k jednotlivým krokům a upravovat jejich provedení.

Obr. 1.1: Metodika CRISP-DM

Jednotlivé fáze podle metodiky CRISP-DM

Porozumění problematice – úkolem této fáze je pochopení cílů úlohy a požadavků na její řešení, hodnotí se rizika a přínosy využití dataminingových metod na řešení konkrétní úlohy. Dále se nastíní návrh řešení úlohy, který bude dál upravován v dalších fázích procesu. Výstupem této fáze je analýza přínosů a nákladů a dynamický plán projektu.

Porozumění datům – základem je sběr dat samotných a první seznámení s daty ve formě deskriptivních statistik (četnost hodnot, minima, maxima, apod.), velmi vhodné je použití různých vizualizačních technik.

Příprava dat – cílem této fáze je sestavit datový soubor, který reprezentuje získaná data a bude dál zpracován analytickými metodami. Využívá se transformace dat, selekce dat, čištění dat a jiné metody pro práci s daty. Transformací dat rozumíme sloučení užitečných dat z více datových souborů do jediné tabulky. Při selekci dat vybíráme atributy, které mají určitou informační hodnotu. Čištění a agregace dat je potřebná pro výpočet odvozených atributů, např. index BMI z atributů hmotnost a výška. Celá tato fáze je obvykle nejpracnější a časově nejnáročnější částí procesu.

Modelování – samotný data mining, slouží k aplikaci zvolené metody na připravený datový soubor a ověření získaných znalostí. Tato část je iterativní, ob-

vykle aplikujeme jednotlivé algoritmy a měníme parametry modelů. V této fázi je velice důležité zaznamenávat provedené analýzy ve formě technických zpráv, aby při další práci nedocházelo ke zbytečnému opakování některých postupů.

Vyhodnocení výsledků – výsledky, resp. znalosti, získané v předchozím kroku je potřeba posoudit z hlediska splnění zadaných cílů, popř. pokusit se o zobecnění výsledků. Podle výsledků rozhodneme o další práci na dané úloze.

Využití výsledků – fáze, kdy získané a ověřené výsledky aplikujeme do praxe např. vytvořením automatického systému pro klasifikaci uživatelů.

Na konci projektu by měla být sepsána závěrečná zpráva, která shrnuje dosažené výsledky a postupy, které byly použity. V některých případech se ještě uvádí revize projektu, což je dokument, který popisuje všechny použité a testované metody a postupy. Tato revizní zpráva slouží pro potřeby data miningové společnosti. Z hlediska významu má největší význam fáze porozumění problému (80 % významu, 20 % času), z hlediska časové náročnější je nejnáročnější fáze přípravy dat (80 % času, 20 % významu). Překvapivě malou částí je samotné modelování (5 % času, 5 % významu).[2, 5, 12]

Podrobný popis této metodiky včetně využití v praxi je volně k dispozici. Tato metodika funguje jako jeden z obecných standardů pro práci v oblasti dobývání znalostí z databází a dále se vyvíjí. V současné době je k dispozici nová verze CRISP-DM 2.0, která byla prezentována začátkem roku 2007 v Londýně.

2 DATA MINING V MEDICÍNĚ

Historické počátky statistického zkoumání dat v medicíně lze zařadit do 19. století. Jsou známy výzkumy, které zkoumaly výskyt a stupeň schizofrenie v závislosti na měsíci narození pacienta nebo výskyt sebevražd v závislosti na fázi měsíce. V této době se jednalo především o lokální výzkumy, které neměly velkou vypovídající hodnotu. Rozvoj statistického bádání v medicíně souvisel s využitím laboratorních metod, nových měřících postupů a uchováváním záznamů pacientů. Základní závislosti v datech byly získány pomocí jednoduchých statistických pozorování, které prováděli sami lékaři s využitím svých poznámek a zkušeností. V dnešní době jsou k dispozici prostředky pro uchování velkého množství dat a jejich následné zpracování. Tyto prostředky nejlépe poskytuje právě data mining. Skoro v každé nemocnici je k dispozici informační systém, který uchovává informace o pacientech a je schopen uložená data exportovat pro další zpracování. Tím jsou splněny dva základní předpoklady pro statistickou úlohu. Máme data, která reprezentují určitý populační výběr a mají statisticky významnou hodnotu. A také máme prostředky pro zpracování těchto dat. Na první pohled se může zdát, že máme ideální podmínky pro použití některé z dataminingových úloh. Realita je ovšem složitější. Data získaná přímo z informačního systému nemocnice jsou v syrové podobě a je proto nutné provést důkladné předzpracování s ohledem na použitý typ úlohy. Data jsou zatížena řadou chyb, které zpravidla nedokážeme odstranit, ale pro práci hrají velkou roli. Lékařství je tak specifickou vědou, že nikdy nemůžeme zkoumat celý tento obor jako celek. Všechny známé studie se zabývají využitím data miningových (popř. statistických) metod pouze v určitém oboru odděleně (např. ortopedie, kardiologie, aj.). A i v těchto oborech se úlohy většinou zaměřují na jeden určitý problém, např. klasifikaci srdeční arytmie. Hledání obecných znalostí je vždy problematické a nově získané znalosti je potřeba podrobit velkému množství testování, než budou uvedeny do praxe.[15, 4]

2.1 Problémy s daty

Problémů s lékařskými daty je mnoho a souvisí s charakterem dat samotných. Data jsou specifická, velice obsáhlá a pro analytiky často nesrozumitelná. Všechny kroky při zpracování dat by měly být konzultovány s expertem - lékařem, aby nedošlo ke ztrátě cenných informací při samotném zpracování. Problémy s daty lze rozdělit do dvou skupin a to na problémy, které souvisí s povahou dat samotných a jejich umístěním v databázích, a na problémy související s etickými a společenskými otázkami.

2.1.1 Problémy s databázemi

Data získaná ve zdravotnictví jsou uchovávána pomocí databázových systémů, které jsou součástí informačních systémů. Většina informačních systémů umožňuje export dat ve formě tabulek, které se stávají vstupem pro data miningové úlohy. Většina dataminingových metod je navržena pro práci s tabulkami. Řádky tabulky reprezentují jednotlivé pacienty, sloupce tabulky reprezentují jednotlivé atributy. Tyto atributy jsou získávány pozorováním, měřením různých údajů nebo jako výstupy z různých vyšetření. Celá tabulka může být velice rozsáhlá, počet pacientů může být v řádu až desítek tisíc, počet atributů několik desítek až stovky. S tím souvisí zásadní problém zpracování databáze jako celku. Data jsou nesourodá, vyšetření a pozorování, která jsou u jednotlivých pacientů provedena, závisí na charakteru případu. Tímto se objevují v databázi chybějící hodnoty, které je potřeba zohlednit. Je možné nahradit chybějící hodnotu např. střední hodnotou, ale v medicíně se tato kompenzace příliš nepoužívá. Vznik dalších chyb je ovlivněn náhodným faktorem – např. chyby při zápisu dat lékařem. Některé atributy obsahují data spojitá (např. hodnota krevního tlaku), jiná data kategoriální (např. kouření – ano/ne). Podle charakteru řešeného problému a použité metody je často nutné tyto atributy transformovat, např. pomocí prahování podle jednoho nebo více parametrů. Pro každou úlohu je potřeba zvolit určitý výběr atributů, ve kterých předpokládáme užitečnou informaci, tento krok se musí opakovat, abychom obdrželi vhodný výběr dat. Dalším problémem může být nesourodost standardů, které daná nemocnice, popř. její oddělení používá. Tyto standardy jsou často specifické pro dané oddělení a pro srovnání s jinými hodnotami je potřeba data transformovat.[2, 4]

Reprezentace hodnot ve formě dvourozměrné tabulky přináší další nevýhodu a tou je nemožnost sledovat časový vývoj hodnot určité veličiny. Musíme proto vědět, zda uvedené hodnoty odpovídají hodnotám, které lékaři naměřili při příjmu pacienta nebo až v průběhu léčby. Časový vývoj hodnot určitého atributu je zpravidla k dispozici, ale jeho začlenění do klasické dvourozměrné tabulky je takřka nemožné. Většinou se musíme spokojit se dvěma údaji, které odpovídají hodnotám daných atributů při příjmu a propuštění pacienta. Jakákoliv práce s takovou informací vyžaduje dobrou znalost problému.[2]

Předzpracování dat je časově nejnáročnější etapou data miningu. V oblasti medicíny je tato náročnost ještě zvýrazněna tím, že každý krok je nutné konzultovat s příslušným expertem, tedy s lékařem, aby nedošlo ke znehodnocení dat nevhodnou úpravou. Je dobré získat určité znalosti o problému, který data popisují. Tyto znalosti mohou usnadnit následnou práci. Je vidět, že můj úkol, zaměřit se na hledání závislostí v databázi z kardiologie je úloha zadaná příliš obecně, a bude nutno se orientovat na určitou, blíže specifikovanou, skupinu pacientů.[12]

2.1.2 Etické a společenské otázky

V dnešní době je důležité zabezpečení osobních dat, která zpracováváme. Při data-miningových úlohách se často pracuje s osobními údaji klientů, je proto nutné snížit riziko úniku informací. V komerční sféře se o toto zabezpečení starají samy firmy, které data mining provádí. Ochrana citlivých údajů je podřízená zákonům daného státu. V České republice je to Zákon č. 101/2000 Sb. o ochraně osobních údajů. Ocituji proto zde některé pasáže tohoto zákona.

> *„Citlivým údajem se rozumí osobní údaj vypovídající o národnostním, rasovém nebo etnickém původu, politických postojích, členství v politických stranách či hnutích nebo odborových či zaměstnaneckých organizacích, náboženství a filozofickém přesvědčení, trestné činnosti, zdravotním stavu a sexuálním životě subjektu údajů.“*

Takto je vymezen pojem citlivý údaj. Je tedy jasné, že všechna zpracovávaná data musí být v souladu s tímto zákonem.

> *„Bez souhlasu subjektu údajů lze osobní údaje zpracovávat pro účely statistické nebo vědecké. Pro tyto účely zpracování je nutno osobní údaje anonymizovat, jakmile je to možné.“*
> *„Správce a zpracovatel jsou povinni přijmout taková opatření, aby nemohlo dojít k neoprávněnému nebo nahodilému přístupu k osobním údajům, k jejich změně, zničení či ztrátě, neoprávněným přenosům, k jejich jinému neoprávněnému zpracování, jakož i k jinému zneužití osobních údajů. Tato povinnost platí i po ukončení zpracování osobních údajů.“*

Před samotnou prací s daty je nutné údaje anonymizovat, v medicíně se tím rozumí zbavit je údajů, podle kterých by mohli být pacienti identifikováni. Jsou to jméno, příjmení, rodné číslo a číslo chorobopisu. A dále je nutné přijmout opatření, aby se data nedostala k rukám třetí osoby.[6, 20]

O těchto problémech a možných řešeních pojednává také práce Julese Bermana, viz [3], zde ale s přihlédnutím k americkým normám a zákonům.

2.2 Techniky data miningu v lékařství

V této části je uveden přehled data miningových technik pro modelování. V současné době existuje mnoho technik, zde je uveden přehled několika z nich se zaměřením na jejich využití v lékařství. Data miningové techniky se dají rozdělit do dvou skupin, na symbolické a subsymbolické metody. Symbolické metody vyhledávají zajímavé

vztahy mezi daty a odhalují skryté struktury v datech. Tyto metody dávají výsledky v podobě logických formulí, které jsou pro koncového uživatele bližší, protože odpovídají lidskému myšlení. Naopak u subsymbolických metod se jedná o učení chápané jako aproximaci matematických funkcí. Výsledkem jsou tedy matematické modely.[13, 15]

2.2.1 Symbolické metody

Rozhodovací pravidla

Pravidla typu Jestliže – potom (If–Then Rules) nazýváme rozhodovací pravidla a jsou nejzákladnějšími pravidly pro klasifikaci. Odpovídají lidskému myšlení, příkladem tohoto pravidla může být věta: „bude-li zima, vezmu si kabát“. První část tohoto pravidla tvoří předpoklad (antecedent), pravou stranu tvoří závěr, který rozhoduje o zařazení případu do určité třídy. Předpoklady můžeme kombinovat s využitím základních logických operací logického součinu, logického součtu a negací. Takto můžeme získat složitější pravidla. Libovolný rozhodovací strom (viz níže) můžeme převést na soubor rozhodovacích pravidel. Rozhodovací pravidla jsou začleněna v algoritmech CN4 a AQ, které jsou dále implementovány v komerčních systémech pro dobývání znalostí. Nejjednodušším algoritmem pro tvorbu rozhodovacích pravidel je algoritmus *pokrývání množin*, známý jako algoritmus AQ. Tento algoritmus hledá pravidla, která pokrývají případy s pozitivní hodnotou cílového atributu. Již pokryté příklady odstraňuje z množiny trénovacích dat. Tento postup se opakuje, dokud v množině trénovacích dat zbývají některé případy. Nevýhodou rozhodovacích pravidel je to, že pracují pouze s kategoriálními daty. Numerické atributy je nutné vhodně diskretizovat (rozdělit do reprezentativních tříd). Tuto diskretizaci je možno provádět ve fázi předzpracování dat nebo přímo při běhu algoritmu. Druhý způsob je využíván systémem CN4 pomocí algoritmu SetBounds(a) pro odhad hraničních bodů, které oddělují jednotlivé třídy. Protože tato pravidla poskytují učení s učitelem, používají se v medicíně pro klasifikaci. Konkrétními aplikacemi mohou být různé soubory pravidel, které se využívají pro podporu diagnostiky v různých oborech lékařství.[2, 15]

Asociační pravidla

Při použití těchto pravidel hledáme vzájemné vztahy mezi atributy. Příkladem může být analýza nákupního vozíku, kdy se zjišťuje, jaké druhy zboží si současně kupují zákazníci v supermarketech. Tato pravidla mají stejně jako rozhodovací pravidla formát Jestliže – potom, zde ale slouží především k predikci nějakého závěru. Levá část pravidla se nazývá předpoklad (antecedent), pravá potom závěr (sukcedent). Základní charakteristikou pravidel jsou dvě odvozené veličiny a to podpora

(support) a spolehlivost (confidence). Podporou rozumíme počet objektů, které splňují předpoklad i závěr. Spolehlivost je podmíněná pravděpodobnost závěru, pokud platí předpoklad. I tato pravidla pracují pouze s diskrétními veličinami, spojité veličiny je opět nutno diskretizovat. Základem všech algoritmů pro hledání asociačních pravidel je generování kombinací hodnot atributů. Počet možných kombinací roste s počtem atributů, to může být problémem u datových souborů, které obsahují velké množství atributů. Asociační pravidla mají schopnost také hledat obecné závěry. Tato pravidla je možno doplnit o pravidla s výjimkami, která poskytují lepší variabilitu pokrytí datového souboru. V medicíně se tato pravidla používají pro identifikaci nových závislostí v datech při dlouhodobějším pozorování a v expertních systémech. Velkou nevýhodou této metody je fakt, že při hledání asociačních pravidel vytváříme všechny kombinace vstupních atributů. Tento fakt dává velkou výpočetní náročnost celého procesu. Pokud budeme uvažovat datový soubor o deseti atributech s binárním rozložením (možné hodnoty 0 nebo 1), máme pro dvouprvkové kombinace předpokladů 144 různých možností výběru. To platí v případě, že známe cílový atribut, který je popsán závěrem pravidla. Pokud jej neznáme, musíme uvažovat, že všechny atributy mohou být atributem cílovým. Tímto se dostáváme na 1440 různých kombinací pro dvouprvkové předpoklady. Nutno podotknout, že počet kombinací je teoretický. Použitím vhodného algoritmu hledání pravidel můžeme tento počet výrazně zmenšit.[1, 2]

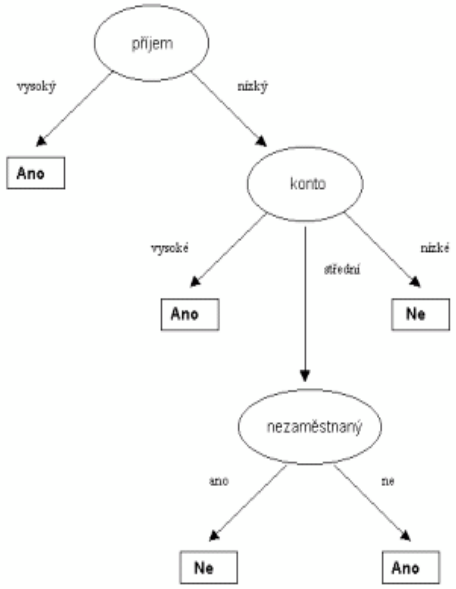
Skryté množiny

Tato analýza se používá k zjištění podobnosti konceptů. Hledá nevnímatelné relace (indiscernibility relation) mezi daty, což jsou objekty se stejnými nebo přibližně stejnými hodnotami atributů. V medicíně se používají při analýze a klasifikaci histologických obrazů, při analýze EEG signálu a v dalších oblastech.[16]

Rozhodovací stromy

Reprezentování znalostí ve formě rozhodovacích stromů je využíváno v řadě oblastí. Příkladem rozhodovacího stromu převedeného na rozhodovací pravidla jsou například klíče pro určování rostlin nebo zvířat. Jejich velkou výhodou je přehlednost a snadná interpretovatelnost znalostí. Rozhodovací strom se skládá z uzlů stromu, což jsou body ve kterých se strom na základě hodnoty některého z atributů větví. Na konci rozhodovacího stromu jsou tzv. listy stromu, podmnožiny, které reprezentují jednotlivé třídy cílového atributu. Příklad rozhodovacího stromu je na obr. 2.1. Na tomto obrázku vidíme, že na základě tří parametrů se vyhodnocuje příslušnost případu k třídě K. Při tvorbě rozhodovacích stromů se nejčastěji postupuje metodou *rozděl a panuj* (divide and conquer). Trénovací data se rozdělují na menší podmnožiny tak, aby v těchto podmnožinách převládaly příklady jedné třídy. Tento

postup se opakuje až na konci máme všechny příklady z trénovací množiny pokryté v jednotlivých podmnožinách stromu. Tento postup bývá často nazýván *top down induction of decision trees* (TDIDT).



Obr. 2.1: Jednoduchý rozhodovací strom

Cílem je nalézt strom, který pokrývá celou množinu trénovacích dat. Tento požadavek naráží na dvě omezení. Prvním je, že i tento postup je určen pro kategoriální data, spojité atributy musíme opět vhodně diskretizovat. Druhým omezením je, že tento algoritmus funguje přesně pro data nezatížená šumem, což medicínská data ze své podstaty nejsou. Musíme volit kompromis, který vytvoří rozhodovací strom za cenu nepřesného zařazení některých případů. Primárně je nutné vybrat určitý atribut jako cílový atribut. U úloh, kdy provádíme klasifikaci (jde tedy o učení s učitelem), je cílový atribut jasně dán. U úloh, kdy se snažíme odhalit závislosti v datech musíme cílový atribut vybrat sami na základě zkušeností, popř. zkoušet více atributů. Samotné větvení stromu je dáno atributem, který nejlépe od sebe odděluje prvky tříd cílového atributu. Nejčastěji se pro toto dělení využívá entropie. Entropie je definována podle vztahu (2.1) jako součet váhovaných pravděpodobností výskytu určitého jevu na množině dat pro všechny možné stavy atributu. Veličina p_t reprezentuje relativní četnost příkladů daného jevu v tabulce.

$$H = -\sum p_t \log_2 p_t \tag{2.1}$$

Pro jeden atribut se entropie vypočítá podle vztahu (2.2). Veličina A_v označuje jednu třídu cílového atributu. Veličina $n_t(A_v)$ je četnost vzájemného výskytu

třídy zvoleného atributu A a cílového atributu, veličina $n(A_v)$ označuje počet prvků pokrytých danou třídou atributu A. Součet se provádí pro všechny možné třídy v atributu A.

$$H = -\sum_{v \in Val(\mathbf{A})} \frac{n_t(A_v)}{n(A_v)} \log_2 \frac{n_t(A_v)}{n(A_v)} \tag{2.2}$$

Dále spočítáme střední entropii $H(A)$ jako vážený součet jednotlivých entropií H pro všechny třídy zvoleného cílového atributu podle vztahu (2.3). Veličina A_v je jedna třída cílového atributu s relativní četností výskytu $n(A_v)$ na množině dat n.

$$H(A) = -\sum \frac{n(A_v)}{n} H(A_v) \tag{2.3}$$

Tento výpočet provedeme pro všechny atributy a pro větvení vybereme atribut, který má nejmenší střední entropii $H(A)$. Tímto jsme vytvořili první uzel rozhodovacího stromu. Dále postupujeme obdobně, ale střední entropie počítáme v rámci jedné větve. Takto postupujeme dokud nejsou všechny případy z trénovací množiny popsány. Pro velké datové soubory můžeme dostat rozsáhlé rozhodovací stromy, existují algoritmy, které provádí tzv. prořezání stromů, kdy za cenu zhoršení klasifikační schopnosti stromu nahradíme uzel přímo listem. S prořezáváním rozhodovacích stromů v oblasti medicíny může být problém, protože zpravidla požadujeme přesnou klasifikaci do jednotlivých tříd. Dalším problémem může být možnost přeučení se stromu na konkrétní případy z trénovací množiny, popř. na šum v datech. Rozhodovací stromy jsou přes svou jednoduchost stále jedním z nejpoužívanějších metod pro data miningové úlohy. V současné době můžeme najít mnoho systému pro tvorbu rozhodovacích stromů. Průkopníkem této metody je možno nazvat Johna Rosse Quinlana, který vytvořil algoritmus ID3, z něhož vychází jeho další algoritmus C4.5 (dnes už ve verzi C5.0). Po zpřístupnění zdrojových kódů těchto algoritmů jsou rozhodovací stromy implementovány do různých komerčních systémů pro dobývání znalostí, například Kepler, Weka nebo Clementine.[2, 15, 8, 19, 11]

Usuzování z případů

Jde o algoritmus, který využívá znalostí získaných předešlým výzkumem ke klasifikaci nových případů. Využívá se velkých databází, které obsahují informace o předchozích případech. Data v těchto databázích musí mít předem definovanou a dobře rozvrženou strukturu pro snadné vyhledávání a rychlé výpočty. Využívá se různých kritérií pro určení podobnosti nového případu s již klasifikovaným a zaznamenaným případem. Pro podobnost se využívají různé metriky, například eukleidovská nebo Hammingova vzdálenost. Pro správný výpočet vzdálenosti je nutná transformace dat. Tato metoda je analogií k metodám vícerozměrné statistické analýzy, kdy

příklady tvoří body v n-rozměrném prostoru. Metoda byla aplikována v lékařství při výzkumech predikce výzkumu rakoviny prsu nebo predikce srdečních chorob ze scintigramů myokardu.[2, 15, 16]

2.2.2 Subsymbolické metody

Umělé neuronové sítě

Tato metoda vychází z matematického modelu neuronových sítí v našem mozku. První matematický model byl vytvořen již v roce 1943 vědci Pittsem a McCullochem. Tento model se nazývá „logický neuron“ a pracoval pouze s vstupními a výstupními hodnotami 0 a 1. Dalším známým modelem je „adaptivní lineární neuron“ Adeline, který byl vytvořen v roce 1960 Widrowem. Vstupem tohoto neuronu jsou numerické hodnoty, které jsou násobeny určitými vahami a podle součtu těchto váhovaných vstupních hodnot se rozhoduje, zda se na výstupu objeví logická nula nebo jednička. V současné době se používají neuronové sítě, kdy je podle určité topologie rozmístěno více neuronů. Umělé neuronové sítě se dají využít jak pro učení s učitelem, tak i pro učení bez učitele. Pro učení s učitelem se nejčastěji využívají vícevrstevné dopředné sítě a Hopfieldovy zpětnovazební sítě, pro učení bez učitele se nejčastěji používají Kohonenovy samoorganizující se mapy nebo metoda SVM (Support Vector Machine). Typickým znakem dopředných neuronových sítí je rozmístění neuronů ve vrstvách. Používá se vrstva vstupních neuronů, několik skrytých vrstev a výstupní vrstva. Pravidlem je, že všechny neurony mezi sousedními vrstvami jsou propojeny vahami. Nastavení hodnot vah se provádí ve fázi učení, využívá se pravidla o zpětném šíření chyby. Kohonenovy mapy jsou tvořeny dvěma vrstvami neuronů, vstupní vrstvou a vrstvou navzájem spojených neuronů. Využívá se principu laterální inhibice, kdy vztahy mezi sousedními neurony v mřížce jsou excitační, vztahy mezi vzdálenějšími neurony jsou inhibiční. Takto postavená struktura má schopnost samoorganizace, tedy shlukování prvků v trénovací množině. Tímto je umožněno učení bez učitele. Metoda SVM využívá datové transformace k převedení datového souboru do tvaru, kdy mohou být jednotlivé třídy lineárně separovatelné. Na rozdíl od využití pravidel jsou vstupem pro umělé neuronové sítě spojité atributy. Případné diskrétní atributy je nutné převést na spojité tak, že každé třídě daného atributu je přiděleno číslo z vhodného intervalu. Využití v medicíně je hlavně při určování prognózy přežití např. v onkologii, ale také v klinické a laboratorní medicíně.[2, 16, 18]

Bayesovská klasifikace

Tyto metody vychází z Bayesovy věty o podmíněné pravděpodobnosti. Ačkoli se jedná o metody pravděpodobnostní, jsou zkoumány v souvislosti se strojovým učením, protože i přes svou jednoduchost a snadnou algoritmizaci jsou velice účinné.

Bayesův vztah (2.4) nám slouží pro výpočet aposteriorní pravděpodobnosti $P(H|E)$, tedy podmíněné pravděpodobnost, že platí hypotéza H při pozorování evidence E. Vychází z apriorní pravděpodobnosti hypotézy $P(H)$, pravděpodobnosti výskytu evidence $P(E)$ a podmíněné pravděpodobnosti $P(E|H)$, která popisuje pozorování evidence E v případě, že platí hypotéza H.

$$P(H|E) = \frac{P(E|H)P(H)}{P(E)} \tag{2.4}$$

Metoda největší věrohodnosti

V reálných případech klasifikace se dostáváme do situace, kdy máme více hypotéz v prostoru hypotéz T a rozhodujeme, která je pro danou evidenci nejpravděpodobnější. V tomto případě přechází jmenovatel Bayesova vztahu do tvaru,

$$\sum_t P(E|H_t)P(H_t)$$

který vyjadřuje úplnou pravděpodobnost evidence E pro všechny hypotézy H_t. Pro všechny hypotézy dostáváme hodnoty aposteriorní pravděpodobnosti a vybíráme hodnotu maximální. Zpravidla nás nezajímá konkrétní hodnota pravděpodobnosti, proto můžeme vztah upravit zanedbáním jmenovatele, který je pro všechny hypotézy stejný. Další úpravou je předpoklad, že všechny hypotézy jsou stejně pravděpodobné a tedy, že nezáleží na jejich pravděpodobnosti $P(H_t)$. Takto jsme obdrželi vztah (2.5), podle kterého určujeme hypotézu s největší věrohodností.

$$H_{ML} = \arg\max P(E|H_t), t \in T \tag{2.5}$$

Naivní bayesovský klasifikátor

Nedostatkem první uvedené metody je, že uvažuje pouze vliv jedné evidence pro posuzování pravděpodobnosti jednotlivých hypotéz. Pokud chceme sledovat vliv více evidencí, a v praxi to tak velmi často bývá, používáme rozšíření metody na Naivní bayesovský klasifikátor (NBK). Vycházíme z předpokladu, že jednotlivé evidence jsou při platnosti dané hypotézy podmíněně nezávislé, díky tomuto zjednodušení se klasifikátor nazývá naivní. Bayesův vztah potom přechází do podoby (2.6). V tomto vztahu jsme opět zanedbali jmenovatele, předpokládáme, že jednotlivé hypotézy jsou stejně pravděpodobné.

$$P(H|E_1, ..., E_K) = \frac{P(H)}{P(E_1, ..., E_K)} \prod_{k=1}^{K} P(E_K|H) \tag{2.6}$$

Pro klasifikaci pomocí této metody budeme opět uvažovat hypotézy z prostoru hypotéz T a vybíráme hypotézu s největší aposteriorní pravděpodobností H_{MAP} podle vztahu (2.7).

$$H_{MAP} = \arg\max P(H_t) \prod_{k=1}^{K} P(E_K|H_t), t \in T \tag{2.7}$$

Všechny veličiny, které pro výpočet potřebujeme, získáme přímo z datového souboru jako pravděpodobnosti určené na základě četností výskytů jednotlivých hodnot. Apriorní pravděpodobnost (2.8) je dána relativní četností výskytu hypotézy v datovém souboru. Podmíněná pravděpodobnost $P(E_K|H_t)$ se vypočítá ze vzájemného výskytu hypotézy H a evidence E v datovém souboru také jako relativní četnost podle vztahu (2.9).

$$P(H_t) = \frac{n_t(H)}{n} \tag{2.8}$$

$$P(E_K|H_t) = \frac{n_t(E_K \wedge H_t)}{n(H_t)} \tag{2.9}$$

Na rozdíl od některých data miningových metod, jako jsou například rozhodovací stromy nebo využití asociačních pravidel, neprohledáváme veškeré možné kombinace evidencí a hypotéz. Tento fakt dává menší výpočetní náročnost, která při zpracování velkých souborů hraje velkou roli. Nevýhodou této metody je, že pokud je ve vztahu 2.7 alespoň jedna podmíněná pravděpodobnost rovna nule, je i výsledná aposteriorní pravděpodobnost nulová. V případě, že je vzájemný výskyt evidence a zvolené hypotézy malý, je výsledná podmíněná pravděpodobnost podhodnocením skutečné pravděpodobnosti. Na odstranění těchto nevýhod se používají Laplaceova korekce při výpočtu apriorní pravděpodobnosti a m-odhad při výpočtu jednotlivých podmíněných pravděpodobností. Laplaceova korekce rozšiřuje výpočet apriorní pravděpodobnosti o přičtení jedničky v čitateli a rozšířením jmenovatele o N_{cl}, které odpovídá počtu tříd. Korigovanou apriorní pravděpodobnost vypočítáme podle vztahu (2.10). M-odhad rozšiřuje výpočet podmíněné pravděpodobnosti o váhování faktorem m podle vztahu 2.11. Veličina f_k odpovídá apriorní pravděpodobnosti evidence E v datovém souboru, veličina m je váhovací faktor, doporučená hodnota je $m = 2$. [16]

$$P(H_t) = \frac{n_t(H) + 1}{n + N_{cl}} \tag{2.10}$$

$$P(E_K|H_t) = \frac{n_t(E_K \wedge H_t) + m \cdot f_k}{n(H_t) + m} \tag{2.11}$$

Tyto dvě metody jsou základní Bayesovské metody používané v data miningových oblastech a můžeme říct, že se staly standardem. V dnešní době je publikováno mnoho metod, které jsou rozšířením těchto metod (např. semi-naivní bayesovský klasifikátor, iterativní klasifikátor, aj.). Dalším rozšířením metod jsou Bayesovské sítě, které jsou obdobou rozhodovacích stromů, ale pro rozhodování vycházejí z pravděpodobností. Bayesovské metody poskytují ve strojovém učení velmi dobré výsledky, v některých případech srovnatelné s umělými neuronovými sítěmi. Nevýhodou může být horší srozumitelnost reprezentace znalostí ve formě pravděpodobností.

Naivní bayesovský klasifikátor je implementován v některých nekomerčních systémech, např. v systému Bayda nebo RoC. V lékařství se užívá hlavně pro diagnostiku nemocí, kdy lze pomocí něj potvrdit či vyvrátit statistické hypotézy. V současné době jsou prezentovány studie, kdy se Bayesovské metody kombinují s rozhodovacími stromy nebo genetickými algoritmy pro zlepšení výsledků daných metod.[2, 7, 15]

Učení založené na instancích

Obdobně jako metoda usuzování z případů vychází tyto metody z míry podobnosti mezi případy. V tomto případě se pro posouzení podobnosti využívá pravidla k-nejbližšího souseda. Při klasifikaci se počítá podobnost mezi novým případem a všemi případy z trénovacích dat. Tento nový případ je zařazen do třídy, která má vzdálenost nejmenší. Jednotlivé příklady si můžeme představit jako body v n-rozměrném prostoru, kde n je rovno počtu atributů a vzdálenost můžeme posuzovat jako eukleidovskou. Předpokladem pro tuto metodu je také prostředek pro ukládání a práci s databází již klasifikovaných případů. Rozšířením této metody je použití tzv. k-d stromů, které představují určitou analogii k rozhodovacím stromům. V listech k-d stromů jsou seznamy podobných případů. Využití v medicíně je hlavně v diagnostice, systémy založené na instancích umožňují rychlejší vytvoření aplikace než v případě klasického znalostního systému.[15, 11]

2.2.3 Transformace dat

Podle potřeb zvolené dataminingové metody se musí data přizpůsobit. Většina dataminingových metod pracuje s diskrétním rozdělením hodnot. Pouze umělé neuronové sítě jsou uzpůsobeny k práci s numerickými atributy. Převod veličin se spojitým rozdělením hodnot na diskrétní se nazývá diskretizace. Nejjednodušším příkladem diskretizace je binarizace, kdy spojitá data přerozdělíme do dvou intervalů. Využívá se opět výpočtu střední entropie, ale v tomto případě počítáme hodnoty střední entropie pro všechny možné dělící intervaly. V praxi by to byl výpočet velkého množství hodnot, proto rozsah veličiny (rozdíl maxima a minima) rozdělíme do několika intervalů a spočítáme entropii v každém z nich. Výpočet se provádí podle vztahu (2.12), kde A_m označuje hodnotu prahu, pro který entropii počítáme.

$$H(A_m) = \frac{n(A < A_m)}{n} H(A(A < A_m) + \frac{n(A > A_m)}{n} H(A(A > A_m) \qquad (2.12)$$

Veličina $H(A(A < A_m)$ je entropie vypočítaná na příkladech, jejichž hodnota je menší než mezní a veličina $H(A(A > A_m)$ entropie na příkladech, jejichž hodnota je větší než mezní. Hodnoty entropie se váhují zlomky, které odpovídají relativním četnostem jednotlivých tříd $n(A < A_m)/n$ a $n(A > A_m)/n$. Jako nejvhodnější mezní

hodnotu určíme tu, která má pro daný atribut nejmenší entropii podle vztahu (2.13).

$$A_m = \arg\min H(A_m), A_m \in < A_{min}; A_{max} > \tag{2.13}$$

Uvedený postup slouží pro určení nejvhodnější mezní hodnoty ze statistického hlediska. V medicíně je tento přístup problematický. Každá transformace dat by měla být konzultována s lékařem - expertem na danou problematiku. V praxi se může stát, že pro atribut systolický krevní tlak obdržíme vzhledem k nějakému cílovému atributu mezní hodnotu 120 mmHg. Tato hodnota může být vzhledem k entropii nejvhodnější, ale z lékařského ohledu je naprosto irelevantní a nevhodná.

Pro použití umělých neuronových sítí musí být všechny atributy spojité. Toho se nejčastěji docílí opět binarizací, kdy pro všechny možné stavy daného atributu vytvoříme nový atribut s binárními hodnotami. Příkladem může být transformace diskrétního atributu kouření, který může nabývat tří stavů: kuřák, nekuřák, stop-kuřák. Pro tuto situaci bude atribut kouření nahrazen třemi novými atributy kuřák, nekuřák, stop-kuřák, které nabývají hodnot 0 nebo 1.[15, 10]

2.3 Ověření správnosti modelů

Velice důležitým krokem v celém procesu dobývání znalostí z databází je zhodnocení správnosti modelů, popř. nově získaných znalostí. Toto zhodnocení se provádí ve spolupráci s expertem na danou oblast. Kromě této expertizy lze provést také hodnocení na základě deskriptivních veličin, které jsou nezávislé na aplikační oblasti daného problému. Vychází se z předpokladu práce s oddělenými daty, kdy před začátkem samotného modelování jsme data rozdělili na data trénovací a testovací. Pro hodnocení nejčastěji vycházíme z tzv. matice záměn (confusion matrix), kdy hodnotíme správnost zařazení (diagnózy) pomocí klasifikace naším systémem a srovnáváme ji se správností zařazení, které je známo z daného testovacího souboru. Matice záměn pro klasifikaci do dvou tříd je znázorněna jako tab. 2.1. TP (správně pozitivní) je počet příkladů, které systém správně zařadil do třídy „+“, FP (falešně pozitivní) je počet příkladů, které systém chybně zařadil do třídy „+“. TN (správně negativní) je počet příkladů, které systém zařadil správně do třídy „-“, FN (falešně negativní) je počet příkladů, které systém chybně zařadil do třídy „-“.

Použití matice záměn je nejjednodušším příkladem testování modelu. Často nás zajímá i o jakou chybu při klasifikaci jde, k zjištění typu chyby musíme použít složitější metody hodnocení. Z matice záměn je možno vypočítat některé jednoduché deskriptivní charakteristiky.

Tab. 2.1: Matice záměn

	Klasifikace systémem	
Správné zařazení	+	-
+	TP	FN
-	FP	TN

Celková správnost zvaná též úspěšnost je charakteristikou toho, jak je daný klasifikační model kvalitní. Vypočítá se podle vztahu 2.14 jako relativní počet správných rozhodnutí systému.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.14}$$

Chyba (Error) se vypočítá obdobně, je to relativní počet nesprávných rozhodnutí systému (2.15)

$$Err = \frac{FP + FN}{TP + TN + FP + FN} \tag{2.15}$$

Dalšími základními charakteristikami jsou senzitivita a specificita. Tyto charakteristiky jsou často používány v medicínských systémech. Senzitivita určuje relativní počet případů popsaných třídou „+“,u kterých jsme správně klasifikovali danou třídu (2.16).

$$Sens = \frac{TP}{TP + FN} \tag{2.16}$$

Specificita určuje u kolika případů popsaných třídou „-“ jsme správně klasifikovali danou třídu (tedy negativní)(2.16).

$$Spec = \frac{TN}{TN + FP} \tag{2.17}$$

Senzitivita a specificita jsou základní deskriptivní charakteristiky pro daný model. V případě, že chceme vyhodnocovat vliv parametru modelu na tyto charakteristiky, používá se tzv. ROC křivka (Receiver Operating Characteristic). Je to graf závislosti hodnoty senzitivity na hodnotě $1 - specificita$. Z matice záměn (2.1) je možné odvodit další deskriptivní veličiny jako např. pozitivní a negativní prediktivní hodnota, poměr falešné negativity nebo prevalence, více viz [17].

Výše uvedené deskriptivní charakteristiky jsou vhodné pro posouzení kvality klasifikačních modelů. V případě, že máme znalosti reprezentované ve formě pravidel, je vhodnější použít hodnocení převzaté z asociačních pravidel. Toto hodnocení vychází z kontingenční tabulky, jejíž podoba pro pravidlo $Ant \Rightarrow Suc$ je uvedena jako tab. 2.2. Řádek „Ant“ odpovídá případům, které jsou pokryty předpokladem pravidla, řádek „$\neg Ant$“ odpovídá případům, které nesplňují předpoklad pravidla.

Sloupec „Suc“ popisuje příklady s platným závěrem pravidla a sloupec „$\neg Suc$“ příklady, které nesplňují závěr. Tato tabulka je obdobou matice záměn, která byla popsána v souvislosti s přesností a chybou modelu.

Tab. 2.2: Kontingenční tabulka

	Suc	$\neg Suc$
Ant	a	b
$\neg Ant$	c	d

Z této tabulky můžeme vypočítat různé charakteristiky pravidel a hodnotit tak získané znalosti. Základními charakteristikami jsou podpora (support) a spolehlivost (confidence).

Podpora je relativní počet objektů splňujících jak předpoklad, tak i závěr. Z kontingenční tabulky se vypočítá pomocí vztahu 2.18.

$$P(Ant \wedge Suc) = \frac{a}{a+b+c+d} \tag{2.18}$$

Spolehlivost je definována jako podmíněná pravděpodobnost závěru pokud platí předpoklad. Je to obdoba senzitivity, která byla uvedena v souvislosti s hodnocením modelu. Výpočet se provádí pomocí vztahu 2.19.

$$P(Suc|Ant) = \frac{a}{a+b} \tag{2.19}$$

Dále nás zpravidla určujeme relativní počty objektů, které splňují předpoklad nebo závěr. V případě počtu objektů, které splňují předpoklad, je to poměr součtu prvků v prvním řádku ku celkovému počtu prvků. Relativní počet objektů, které splňují závěr, je to poměr součtu prvků v prvním sloupci ku celkovému počtu prvků. Tyto veličiny vypočítáme pomocí vztahů (2.20) a (2.21).

$$P(Ant) = \frac{a+b}{a+b+c+d} \tag{2.20}$$

$$P(Suc) = \frac{a+c}{a+b+c+d} \tag{2.21}$$

Kombinovanou charakteristikou, která vyjadřuje sílu pravidla, je kvalita. Ta je definována jako vážený součet spolehlivosti a pokrytí podle vztahu (2.22), kde veličiny w_1 a w_2 se volí tak, aby jejich součet byl jedna. V praxi se používá rovnoměrné rozdělení ($w_1 = w_2 = 0,5$) nebo $w_1 = 0,8$ a $w_2 = 0,2$.

$$Q = w_1 \frac{a}{a+b} + w_2 \frac{a}{a+c} \tag{2.22}$$

Ačkoliv jsou tyto charakteristiky používány především v souvislosti s asociačními pravidly, lze je použít i na rozhodovací pravidla a závěry získané z výpočtu naivního bayesovského klasifikátoru.[2, 16]

3 REALIZACE PROCESU DOBÝVÁNÍ ZNALOSTÍ Z DATABÁZÍ

3.1 Registr IKK FN Brno

Na Interní kardiologické klinice Fakultní nemocnice Brno Bohunice (dále jen IKK) funguje tzv. registr IKK. Je to databáze všech pacientů, kteří byli hospitalizováni na tomto oddělení. Tento registr vychází z informačního systému nemocnice, ze kterého také přebírá některá data. Celý registr je realizován na SQL serveru a jednotlivé položky jsou zapisovány samotnými lékaři. Tento soubor obsahuje široké spektrum atributů zahrnujících např. anamnézu pacienta, užívané léky, některé laboratorní hodnoty, aj.. Zápis dat je standardizován díky aplikaci na SQL serveru, ale data samotná nepodléhají žádné systémové kontrole, mohou se objevit chyby vzniklé při zápisu, jako překlepy. Díky zápisu v informačním systému lze odhalit pouze některé chyby (např. překlep při zápisu diskrétních veličin), chyby při zapisování atributů se spojitým rozdělením nelze odhalit ani zpětně dohledat. Databáze zatím neobsahuje hodnoty laboratorních vyšetření, která jsou k dispozici jako výstupní soubory přímo z klinických laboratoří. Zahrnutí do stávající databáze je problematické, protože databáze jako taková neumí pracovat s časovým vývojem hodnot. Další problém by byl s faktem, že laboratorní vyšetření se provádí podle stavu pacienta a pro každého pacienta se provádí vyšetření jiná. Tím by databáze zvětšila svůj objem a stala by se nepřehlednou. Problémem této databáze je také přítomnost chybějících údajů. Tyto údaje nebyly zapsány buď z nedbalosti lékaře, nebo v častějším případě z důvodu, že nebyly relevantní pro stav a diagnózu daného pacienta. V dataminingové praxi je mnoho metod pro doplnění chybějících údajů na základě střední nebo nejčastější hodnoty. V medicínských datech se kvůli jedinečnosti každého záznamu příliš nepoužívají.

Registr obsahuje celkem 124 atributů, z nichž 26 má spojité rozdělení hodnot (krevní tlak), 95 rozdělení diskrétní (užívání skupin léků). S tímto faktem je třeba také počítat při výběru data miningové metody — některé metody pracují pouze s diskrétními a jiné pouze se spojitými daty. Registr obsahuje záznamy celkem 16 370 pacientů zaznamenané od počátku roku 2005 doposud. Data byla exportována jako tabulka programu Microsoft Excel (typ souboru xls). Anonymizace dat byla provedena v programu Microsoft Excel tak, aby datový soubor byl v souladu se zněním Zákona č. 101/2000 Sb. o ochraně osobních údajů. Diagnózy pacientů jsou od února 2006 kódovány pomocí standardního číselníku MKN10 (mezinárodní klasifikace nemocí). Tento číselník je implementován také ve většině informačních systémů, které se používají ve zdravotnictví. Pro velkou obsáhlost tohoto číselníku – 12 000 diagnóz

– nezahrnuji data o diagnóze do této práce.

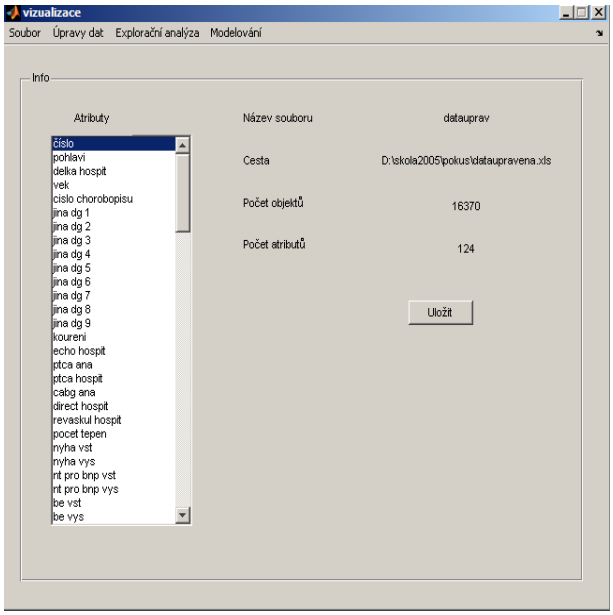
3.2 Předzpracování dat

Pro práci s daty byl zvolen počítačový program Matlab, který je k dispozici v počítačových laboratořích školy. Hlavním důvodem je jednoduchost a rychlost práce s maticemi. Všechny popsané aplikace byly vytvořeny v prostředí GUI (Graphics User Interface), které je součástí programu Matlab. V rámci této práce je postupováno podle metodiky CRISP-DM, která byla popsána.

3.2.1 Porozumění datům

Úkolem této fáze je sběr dat a seznámení se s charakterem dat samotných. Sběr dat je prováděn přímo na IKK. Před samotnou analýzou datového souboru bylo potřeba provést některé úpravy datového souboru. Data byla vyexportována ve formě tabulky programu Microsoft Excel, základní úpravy byly tedy prováděny v tomto programu. Hlavička tabulky, která obsahuje názvy jednotlivých atributů byla přesunuta do Listu2 tak, aby v Listu1 zůstaly pouze hodnoty jednotlivých atributů. Do Listu2 byla také pro každý atribut doplněna informace o charakteru rozdělení. Hodnotou 0 byly označeny atributy jejichž rozdělení neodpovídá ani spojitému ani diskrétnímu, hodnotou 1 atributy se spojitým rozdělením a hodnotou 2 atributy s diskrétním rozdělením. Do Listu3 byly umístěny informace o hodnotách jednotlivých atributů. Pro spojité veličiny to byla jednotka dané veličiny, pro diskrétní veličiny to byl výčet hodnot, které může atribut nabývat. Při dodržení této struktury kódování je možné použít vyvinuté aplikace k analýze libovolné tabulky. Pro načítání dat v programu Matlab byla vytvořena funkce *otevritxls*, která respektuje tuto strukturu dat. Po spuštění funkce se otevře dialog pro výběr datového souboru ve formátu xls. Výstupními proměnnými jsou informace o souboru (cesta, název souboru), data samotná, vektor atributů a dodatečné informace o souboru. Samotné načtení tabulky je provedeno pomocí funkce Matlabu *xlsread*. Pro rychlejší práci s rozsáhlým datovým souborem byla vytvořena aplikace s názvem *vizualizace*, ze které jsou spouštěny všechny ostatní aplikace. Tato aplikace realizuje načtení datového souboru a uložení proměnných do souboru *datafile.mat*, který slouží ostatním aplikacím jako zdroj dat. Vzhled této aplikace je na obr. 3.1.

V rámci prvotního seznámení s daty byla provedena analýza chybějících údajů v databázi. Hodnoty, které nejsou zapsané, jsou po načtení v Matlabu kódovány hodnotou 0. Procento chybějících hodnot je různé. Je to dáno různými skutečnostmi. Veličiny se spojitým rozdělením nejsou do registru zapisovány od roku 2008, u těchto



Obr. 3.1: Vzhled nadřazené aplikace

atributů je podíl nulových hodnot více než 50 %. V diskrétních atributech je podíl nulových hodnot průměrně 10 %. Tuto skutečnost musíme zohlednit při výběru dat a volbě vhodné dataminingové metody.

3.2.2 Vizualizace dat

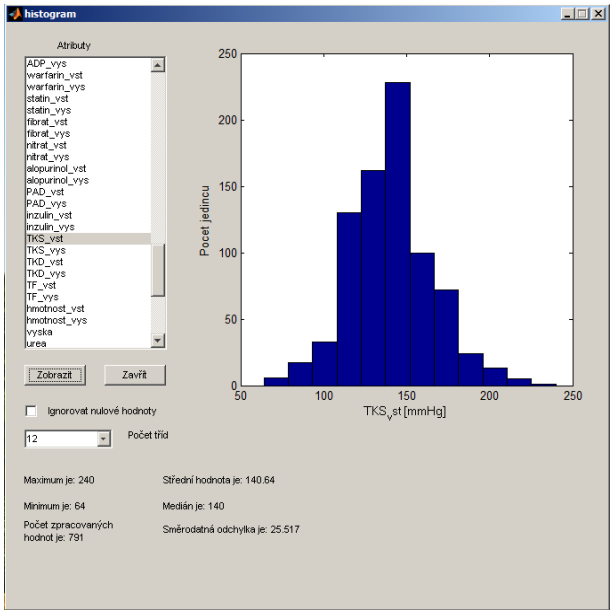
Vizualizace dat, tedy zobrazení některých deskriptivních charakteristik, je základním nástrojem pro prvotní analýzu dat. Vizualizace dat je také základem statistického oboru explorační analýza dat, který zkoumá data ne na základě statistických hypotéz, ale na základě různých vizualizací, viz [10]. Z tohoto důvodu byla vytvořena aplikace, která počítají a zobrazují základní parametry pro jednotlivé atributy. Byly vytvořeny aplikace pro zobrazení histogramu, histogramu s parametrem a klasického xy zobrazení.

Zobrazení histogramu

Histogram je nejznámější způsob zobrazení hodnot jedné proměnné. Na osu X jsou vynášeny hodnoty proměnné, na osu Y absolutní nebo relativní četnosti jednotlivých tříd hodnot. V případě atributů s diskrétním rozdělením hodnot jsou na ose X zobrazeny jednotlivé třídy. V případě atributů se spojitým rozdělením jsou na ose X zobrazeny intervaly, které pokrývají celý rozsah hodnot. Počet těchto intervalů je standardně nastaven na 10, ale podle potřeby je možné ručně změnit počet intervalů

v rozsahu od 2 do 15. Reprezentace dat histogramem nám dává informaci o přítomnosti odlehlých hodnot, o charakteru rozdělení dané proměnné, o přítomnosti shluků hodnot. U spojitých veličin často prokládáme histogram křivkou hustoty některého statistického rozdělení (normálního, lognormálního, binomického). Podle tvaru histogramu a míry shody s touto ideální křivkou můžeme určit další statistické veličiny jako jsou koeficienty špičatosti a šikmosti. [10]

Vytvořená aplikace zobrazuje histogram zvoleného atributu. Zobrazení histogramu je realizováno pomocí funkce *hist*, která má dva vstupní parametry – vektor dat, jejichž histogram chceme zobrazit a počet intervalů na ose x. Také je možno zvolit, zda chceme nebo nechceme zobrazovat v histogramu nulové hodnoty. Tyto hodnoty mohou mít vliv na tvar histogramu. Odstranění nulových hodnot bylo realizováno pomocnou funkcí. Pro zvolený atribut se kromě zobrazení histogramu počítají také některé statistické veličiny. Pro výpočet těchto veličin byly použity standardní funkce programu Matlab. Pro výpočet střední hodnoty je použita funkce *mean(y)*, pro medián funkce *med(y)*, pro maximum *max(y)*, pro minimum *min(y)* a pro směrodatnou odchylku funkce *std(y)*. Vzhled aplikace s příkladem zobrazení pro systolický krevní tlak je na 3.2. Byly ošetřeny stavy zobrazení histogramů u atributů, které nemají spojité ani diskrétní rozdělení.

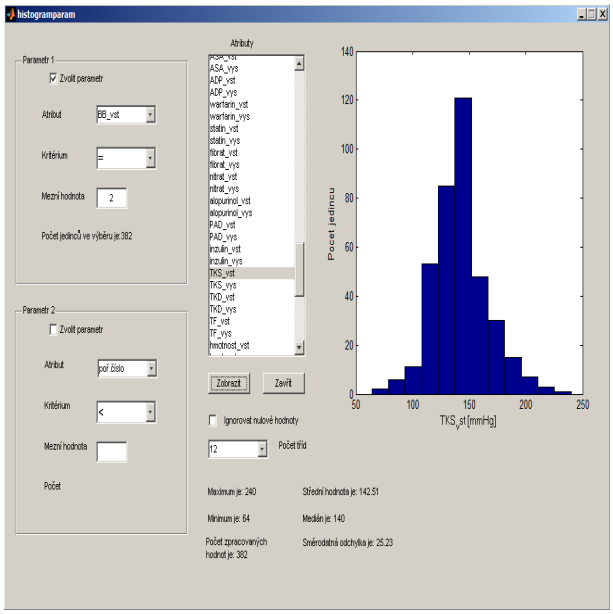


Obr. 3.2: Zobrazení histogramu

Zobrazení histogramu s parametrem

Zobrazení histogramu nám dává základní informaci o zvoleném atributu. Pro zís-

kání více informací o datovém souboru byla vytvořena aplikace, která zobrazuje histogram s volitelnými parametry. Volba parametru nám dává větší možnosti pro analýzu dat. Můžeme například realizovat zobrazení histogramu veličiny systolický krevní tlak pro pacienty, kteří užívají léky ze skupiny betablokátorů. Aplikace pracuje stejně jako předchozí, jen je pozměněn vektor hodnot, jejichž četnosti zobrazujeme. Vektor hodnot je upraven podle zvoleného parametru. Jako parametr se zadává název atributu, logické kritérium a mezní hodnota. Logickým kritériem jsou základní relace menší než, větší než a rovno. Pro úpravu vektoru hodnot byla vytvořena pomocná funkce *vyberdat*. Vstupními veličinami jsou zvolený datový soubor, název parametru, logické kritérium a mezní hodnota. Výstupem této funkce je datový soubor, který odpovídá zparametrizovaným datům. Stejně jako předchozí aplikace jsou i zde vypočteny a zobrazeny základní statistické veličiny. Také je provedena možnost volby zobrazení nulových hodnot. Důležitým parametrem je zde počet zpracovaných dat, nevhodnou volbou parametru můžeme dostat histogram, který nemá statisticky významnou hodnotu. Je možné zvolit dva různé parametry. Vzhled aplikace s příkladem zobrazení histogramu pro systolický krevní tlak u pacientů, kteří užívají betablokátory je na 3.3.

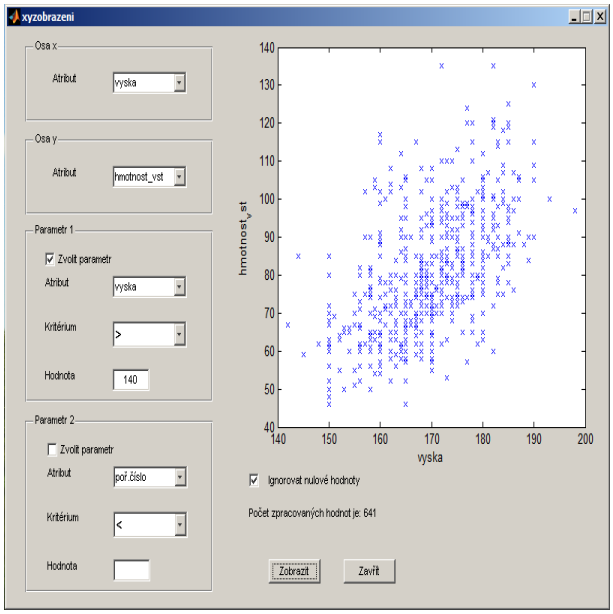


Obr. 3.3: Zobrazení histogramu s parametrem

XY zobrazení

Pomocí tohoto zobrazení můžeme vyhodnotit jednoduché závislosti mezi jednotlivými atributy. Je to základní zobrazení dvourozměrných dat, které zobrazuje hod-

noty jednoho atributu jako funkci atributu druhého. Příkladem takového zobrazení může být zobrazení hodnot systolického krevního tlaku v závislosti na hmotnosti pacienta. Toto zobrazení má smysl pouze pro dvojici spojitých atributů. V případě diskrétních atributů dostaneme pouze shluky hodnot, které nemají vypovídací hodnotu. Zobrazuje se graf, jehož body odpovídají jednotlivým pacientům, kteří jsou popsáni dvojicí hodnot zvolených atributů. Podle charakteru rozložení bodů můžeme určit, zda je mezi zvolenými atributy nějaká korelace nebo není. Vyhodnocení korelací je komplikovanější, zabývají se jím statistické obory korelační a regresní analýza, více viz [10]. Zobrazení grafu je realizováno pomocí funkce Matlabu *plot(x,y)*, které vykreslí graf závislosti hodnot vektoru y na hodnotách vektoru x. Také pro toto zobrazení je umožněn výběr až dvou volitelných parametrů. Princip parametrizace dat je stejný jako u zobrazení histogramu s parametrem. Obdobně jako v předešlých aplikacích je možno zvolit zobrazení s nebo bez nulových hodnot. Vzhled aplikace s příkladem XY zobrazení pro závislost hmotnosti pacienta na jeho výšce je na obr. 3.4.



Obr. 3.4: XY zobrazení

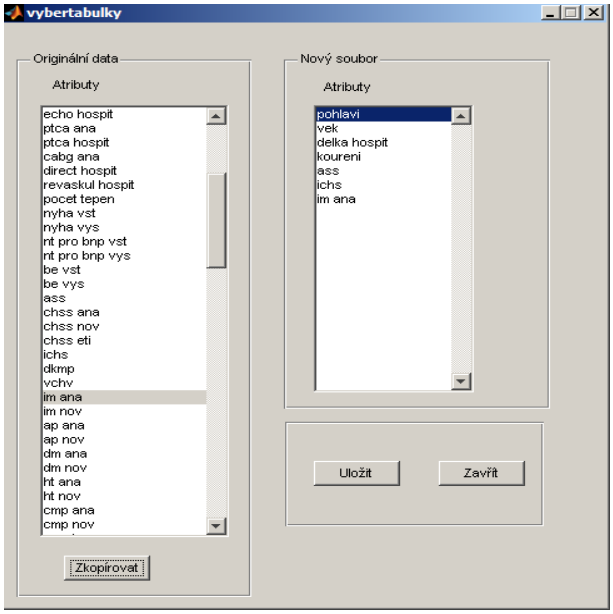
3.2.3 Příprava dat

Úkolem této fáze je sestavit datový soubor, který dobře reprezentuje získaná data a bude ho možno dál zpracovat dataminingovými metodami. Pro účely přípravy dat byly vytvořeny tři aplikace, které realizovaly jednotlivé kroky přípravy dat: selekci, čištění dat a transformaci. Selekcí rozumíme výběr atributů, které zahrneme

do další práce. Čištění dat je proces, kdy podle logického kritéria vybíráme pro další práci pouze data konkrétních hodnot. Transformaci v tomto případě chápeme jako vytvoření binární tabulky podle zvoleného kritéria. Tyto tři části jsou obvykle realizovány v tomto daném pořadí.

Selekce

Pro selekci byla vytvořena aplikace, která realizovala prostý výběr sloupců tabulky pro další práci. Samotná činnost je realizována jako kopírování zvolených atributů do nového datového souboru, který je posléze uložen opět jako tabulka programu Microsoft Excel pro další práci. Struktura datového souboru zůstává stejná. Vzhled aplikace je uveden na obr. 3.5

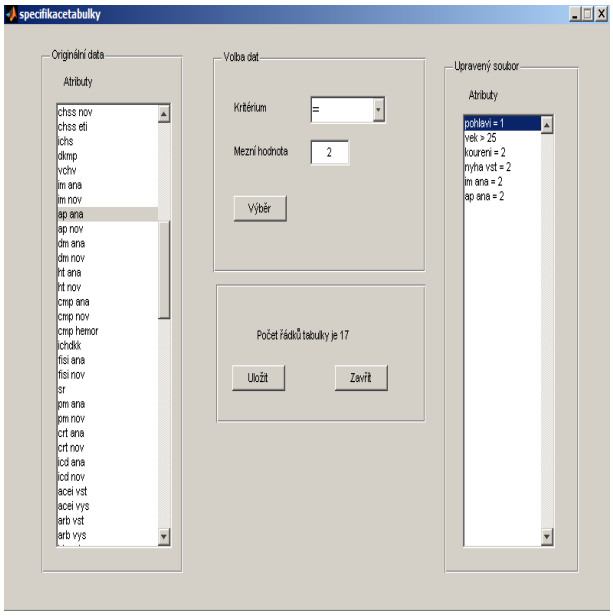


Obr. 3.5: Výběr atributů tabulky

Čištění dat

Čištění dat slouží pro odstranění irelevantních hodnot v rámci jednotlivých atributů. Výběr dat je realizován pomocí funkce *vyberdat*, která pro zvolený atribut, kritérium a mezní hodnota provede ošetření hodnot dat. Hodnoty, které splňují logickou podmínkou jsou zahrnuty do nového datového souboru, hodnoty, které nesplňují jsou vymazány. Aplikace je navržena tak, aby bylo možno provádět opakované čištění hodnot pro zvolený atribut. Výstupem této aplikace je tabulka se stejným počtem atributů jako vstupní, ale s provedeným ořezáním některých hodnot. Tabulka je opět uložena jako tabulka programu Microsoft Excel se všemi informacemi

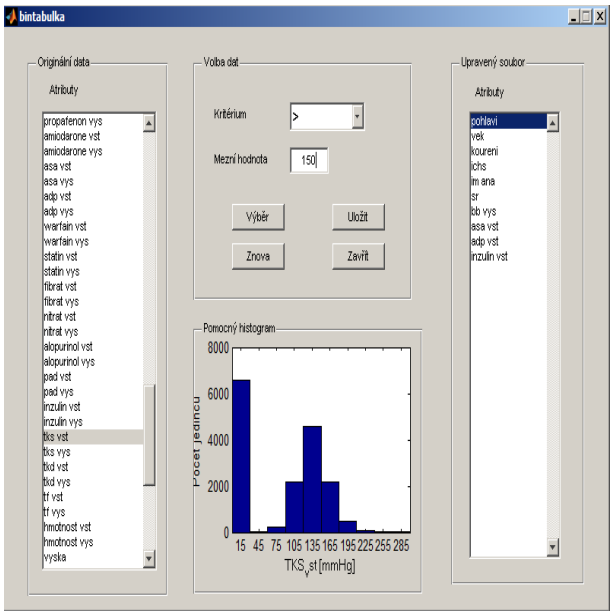
pro další práci. Příklad využití této aplikace pro tvorbu konzistentní tabulky, která neobsahuje nulové hodnoty, je uveden na obr. 3.6.



Obr. 3.6: Specifikace datového souboru

Tvorba binární tabulky

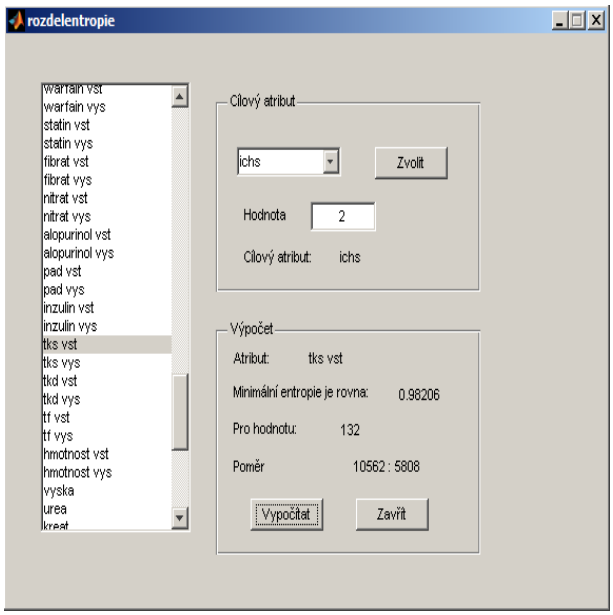
S ohledem na zvolené metody pro další práci je nutno datový soubor upravit do podoby binární tabulky. Binární tabulkou rozumíme tabulku, která nabývá hodnot z dvouprvkové množiny (0,1). Pro tuto úpravu byla vytvořena aplikace, která podle platnosti zvoleného logického kritéria rozhoduje o zařazení do třídy 0 nebo 1. Pro realizaci byla napsána funkce *binvyber*, která pro všechny hodnoty daného atributu realizuje srovnání s mezní hodnotou a zařazení do třídy 0 nebo 1. Vstupem této funkce je datový soubor, zvolený atribut, který upravujeme, logické kritérium a mezní hodnota. Výstupem této funkce je vektor hodnot příslušného atributu s hodnotami 0,1. Pro tvorbu binární tabulky voláme tuto funkci opakovaně pro jednotlivé atributy. Pro orientaci v rozložení hodnot zvoleného atributu je zobrazena zmenšená verze histogramu. Výstup je uložen jako tabulka aplikace Microsoft Excel, která má v Listu3 zapsána pro jednotlivé atributy logická kritéria, podle kterých byla data transformována. Generování ostatních tabulek pro program Microsoft Excel je stejné jako v předcházejících případech. Vzhled aplikace pro konkrétní binární tabulku je na obr. 3.7.



Obr. 3.7: Tvorba binární tabulky

Volba vhodného prahu

Pro většinu metod jsou vstupem atributy s diskrétním rozdělením hodnot. Pro spojité musíme vhodně zvolit práh podle kterého rozhodneme. Pro volbu prahu s ohledem na minimální střední entropii byla navržena aplikace, která pro cílový atribut a vybraný atribut vypočítá nejmenší střední entropii. Problémem ale často je, že v této fázi nevíme, který atribut bude ve skutečnosti cílový. Proto je možná také volba na základě empirie. Pro rozhodnutí na základě entropie byla vytvořena funkce *vypentrop*, která pro zvolenou kombinaci atributů počítá rozdělení entropie podle vztahu 2.13. Vstupem je datový soubor, čísla dvou zvolených atributů, hodnota cílového atributu, která nás zajímá a typ veličiny. Výstupem je potom hodnota nejmenší střední entropie a vypočtená mez. Pro diskrétní veličiny počítá funkce hodnoty entropie pro všechny možné hodnoty. Pro spojité veličiny provádí funkce rozdělení rozsahu hodnot do sto dílčích intervalů a v každém vypočítá hodnotu entropie. Jako nejlepší mez pro rozdělení je doporučen střed intervalu, který má nejmenší entropii. V aplikaci se zvolí cílový a zkoumaný atribut a hodnota cílového atributu, který nás zajímá a zobrazí se nám hodnota minimální entropie a hodnota veličiny, kdy je entropie minimální. Dále je zobrazen také poměr hodnot po rozdělení což může být v některých případech důležité. Tato aplikace slouží pro lepší odhad vhodného mezního bodu pro tvorbu binární tabulky podle vypočtené entropie, vzhled aplikace je na obr. 3.8. Při práci s medicínskými daty je to ale poněkud problematické, jak bylo uvedeno výše.



Obr. 3.8: Aplikace pro výběr vhodného prahu

3.3 Modelování

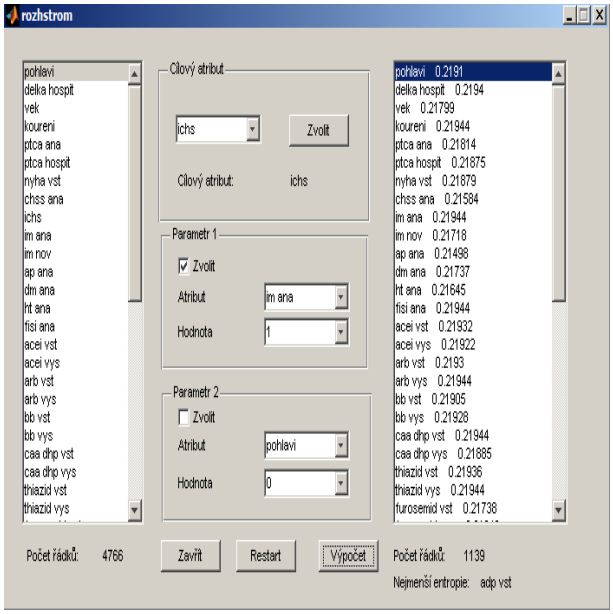
Další částí procesu dobývání znalostí z databází je modelování, tedy realizace jednotlivých metod. Byly realizovány rozhodovací stromy, metoda největší věrohodnosti, naivní bayesovský klasifikátor a model dopředné neuronové sítě. Všechny popsané metody byly realizovány v prostředí Matlab formou GUI (Graphics User Interface) nebo skriptů. Součástí této kapitoly je také ověření zpracovaných modelů na testovacím příkladu.

3.3.1 Rozhodovací stromy

Rozhodovací stromy byly vybrány pro možnost snadného převedení stromu na rozhodovací pravidla a tedy k analýze závislostí. V rámci této práce je postupováno podle výše popsaného algoritmu *top down induction of decision trees* s rozhodováním na základě střední entropie. Pro tuto práci byla navržena aplikace, která počítá pro zvolený cílový atribut střední entropii pro všechny ostatní atributy. Pro tento výpočet je možné zvolit dva různé atributy, které realizují postup v rozhodovacím stromu do druhé úrovně se všemi možnými kombinacemi. Celý rozhodovací strom není realizován, protože naším zájmem je hledat základní závislosti v datech platné na celém datovém souboru.

Pro výpočet byla napsána funkce *entropie*, která realizuje výpočet střední entropie podle vztahu 2.3. Vstupem této funkce jsou data ve formě binární tabulky, cílový atribut a zvolený atribut. Výstupem je hodnota střední entropie pro danou kombi-

naci atributů. Tato funkce je volána pro všechny atributy mimo cílového a výsledky jsou zobrazeny. Atribut s nejmenší entropií je následně vypsán. Pro specifikaci datového souboru je použita funkce *vyberdat*. Pro lepší orientaci v datovém souboru, je počet jedinců v dané specifikaci vypsán. Samotná aplikace před výpočtem ověřuje, zda datový soubor je ve formě binární tabulky. K tomuto ověření je napsána funkce *isbinartab*, která pro zadaný datový soubor zkoumá, zda obsahuje pouze nuly nebo jedničky. Samotný algoritmus je jednoduchý, v cyklu se prochází všechny pole datového souboru a v případě, že daná buňka neobsahuje hodnotu nula nebo jedna, je tabulka označena za nevyhovující. Na obrázku 3.9 je ukázka výpočtu konkrétního rozhodovacího stromu s cílovým atributem ICHS (ischemická choroba srdeční).



Obr. 3.9: Výpočet rozhodovacího stromu

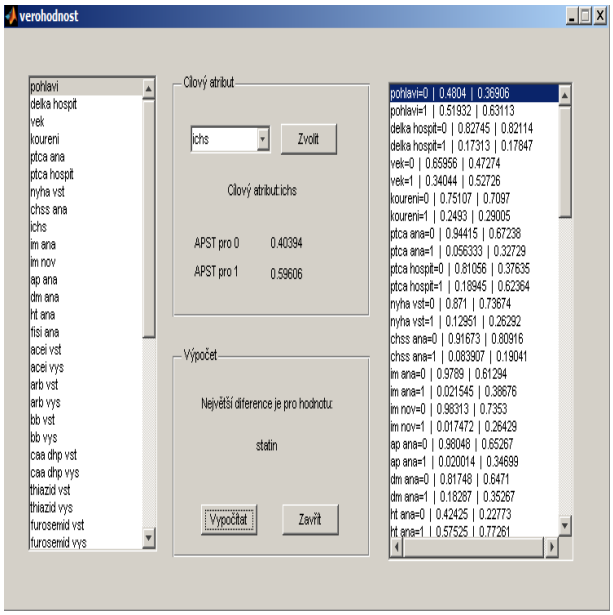
3.3.2 Dopředná neuronová síť

Umělé neuronové sítě se v procesu dobývání znalostí z databází používají pro klasifikaci nebo predikci. Pro hledání závislostí nejsou příliš vhodné kvůli obtížné analýze chování sítě ve skrytých vrstvách. Tato metoda je realizována pro klasifikaci dat do jednoho konkrétního atributu. Byla použita dopředná neuronová síť se zpětným šířením chyby, která obsahovala dvě skryté vrstvy. Tato metoda je realizována ve formě skriptu. Datový soubor přebírá z nadřazené aplikace *vizualizace* samotná data a přidružené informace. Dále jsou data transformována do podoby vhodné pro použití neuronové sítě. Jsou rozdělena na trénovací a testovací, počty hodnot v jednotlivých souborech jsou voleny rovnoměrně. Dále je datový soubor rozdělen na data

vstupní a data výstupní, která obsahují hodnoty v cílovém atributu. Realizace samotné neuronové sítě je pomocí funkcí toolboxu Neural Networks. Přednastaveno je 1000 epoch učení sítě. Pro hodnocení kvality neuronové sítě je vypočtena hodnota správnosti klasifikace, která je dána poměrem mezi počtem správně klasifikovaných evidencí a celkovým počtem evidencí. Správnost klasifikace je určována na trénovacích i testovacích datech. Protože výstup neuronové sítě je spojitý na intervalu $\langle 0; 1\rangle$ je správnost klasifikace hodnocena podle kritéria, které má předdefinovanou hodnotu 0,2. To znamená, že pokud je rozdíl mezi správnou klasifikací a klasifikací pomocí neuronové sítě menší než 0,2, je označena klasifikace za správnou.

3.3.3 Metoda největší věrohodnosti

První z realizovaných bayesovských metod je metoda největší věrohodnosti, která slouží pro určení třídy, která nastane s větší pravděpodobností. Pro výpočet největší věrohodnosti byla vytvořena aplikace, která pro zvolený cílový atribut počítá podmíněné pravděpodobnosti podle vztahu 2.5. Jsou vypočítány podmíněné pravděpodobnosti pro všechny atributy v kombinaci s atributem cílovým. Pro výpočet podmíněné pravděpodobnosti byla vytvořena funkce *podmpst*, která počítá se zahrnutím m-odhadu podle vztahu 2.11. Vstupem této funkce jsou datový soubor, indexy zvoleného a cílového atributu a hodnota zvoleného atributu. Výstupem jsou dvě podmíněné pravděpodobnosti pro oba stavy cílového atributu. Jednotlivé veličiny jsou získány jako relativní četnosti výskytu evidence E v datech za platnosti nebo neplatnosti hypotézy H. Protože se pohybujeme v prostoru atributů s binárním rozdělením, výsledkem jsou pro každou možnou hodnotu atributu dvě podmíněné pravděpodobnosti, které odpovídají dvěma možným stavům cílového atributu. V tomto případě vyhodnocujeme absolutní hodnotu rozdílu mezi podmíněnými pravděpodobnostmi, které odpovídají platnosti a neplatnosti cílového atributu. V případě atributů, které dobře oddělují prvky jednotlivých tříd cílového atributu, bude tento rozdíl velký. V případě, že zvolený atribut nemá vliv na hodnotu cílového (není mezi nimi relace), bude tento rozdíl malý. V rámci aplikace je v cyklu volána funkce *podmpst* pro každou možnou kombinaci atributů s jedním konkrétním cílovým atributem. Název atributu, který má největší rozdíl mezi pravděpodobnostmi je vypsán. Problém může nastat u atributů, jejichž apriorní pravděpodobnost výskytu je příliš nízká. V tomto případě jsou získané podmíněné pravděpodobnosti silně ovlivněny původní apriorní pravděpodobností a hodnoty pro oba stavy cílového atributu jsou velice podobné. Na obr. 3.10 je ukázka výpočtu podmíněných pravděpodobností pro cílový atribut ICHS.



Obr. 3.10: Aplikace pro výpočet metody největší věrohodnosti

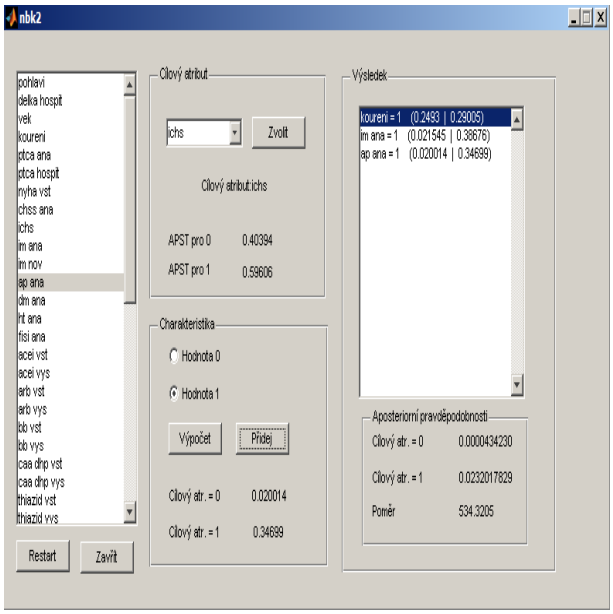
3.3.4 Naivní bayesovský klasifikátor

Naivní bayesovský klasifikátor je spolu s umělými neuronovými sítěmi jednou z nejpoužívanějších subsymbolických metod v dataminingu. Důvodem je, že v rámci dobývání znalostí z databází, dává velmi dobré výsledky. Pro výpočet naivního bayesovského klasifikátoru byla vytvořena aplikace, která pro zvolený cílový atribut a zvolenou kombinaci evidencí určí aposteriorní pravděpodobnost podle vztahu 2.7. Výpočet dílčích pravděpodobností je prováděn se zahrnutím Laplaceovy korekce a m-odhadu. Pro výpočet apriorní pravděpodobnosti byla napsána funkce *aprpst*, která provádí výpočet se zahrnutím Laplaceovy korekce podle vztahu 2.10. Vstupem této funkce jsou samotná data ve formě binární tabulky a číslo atributu, pro který pravděpodobnost počítáme. Výstupem této funkce jsou dvě apriorní pravděpodobnosti, odpovídající platnosti a neplatnosti zvoleného atributu. Pro výpočet podmíněné pravděpodobnosti byla vytvořena funkce *podmpst*, která byla popsána výše.

Samotná algoritmizace naivního bayesovského klasifikátoru je jednoduchá, protože výsledná aposteriorní pravděpodobnost je dána součinem apriorní pravděpodobnosti pro hypotézu a příspěvky jednotlivých pozorovaných evidencí. Pro zvolený cílový atribut vypočítá aplikace apriorní pravděpodobnosti pro obě třídy. Dále se přidávají jednotlivé evidence, kterými je daná apriorní pravděpodobnost násobena. Evidence obsahuje zvolený atribut a hodnotu, kterou zvolený atribut nabývá, pro usnadnění práce je před samotným přidáním evidence hodnota podmíněných pravděpodobností, které přísluší evidenci, vyčíslena. Tímto jednoduchým způsobem mů-

žeme zkoumat evidence libovolné délky. Výsledné aposteriorní pravděpodobnosti se zobrazují společně s poměrem, který je dán podílem větší a menší hodnoty aposteriorní pravděpodobnosti. Tento poměr nám udává číslo, kolikrát je větší pravděpodobnost, že nastane stav s větší aposteriorní pravděpodobností. Ačkoliv je do výpočtu naivního bayesovského klasifikátoru započítána Laplaceova korekce i m-odhad, v případě, že apriorní pravděpodobnost výskytu hypotézy nebo evidence je nízká, dochází ke značnému zkreslení výstupu a výsledný poměr obou pravděpodobností roste nade všechny meze.

Výpočet poměru mezi jednotlivými aposteriorními pravděpodobnostmi slouží jako ukazatel, zda je vhodnou kombinaci evidencí a cílového atributu možno převést na pravidlo. Každé takto získané pravidlo musí být otestováno a vypočteny jeho deskriptivní charakteristiky. Pro ověření získaného pravidla může posloužit histogram s parametrem, popř. výpočet některých deskriptivních veličin, např. podpora, spolehlivost, senzitivita nebo specificita. Vzhled aplikace je na obr. 3.11, v tomto konkrétním případě se počítají aposteriorní pravděpodobnosti pro evidenci *Kouření=ano*, $IM\,ana{=}ano$ a $AP\,ana{=}ano$ vzhledem k cílovému atributu ICHS. Pravděpodobnější z obou hypotéz je hypotéza *ICHS=ano*.



Obr. 3.11: Aplikace pro výpočet naivního bayesovského klasifikátoru

3.3.5 Automatické hledání závislostí pomocí NBK

Pro potřeby IKK Fakultní nemocnice Brno bylo potřeba vytvořit aplikaci, která provádí automatické hledání závislostí v datech. Díky dobrým výsledkům a snadné

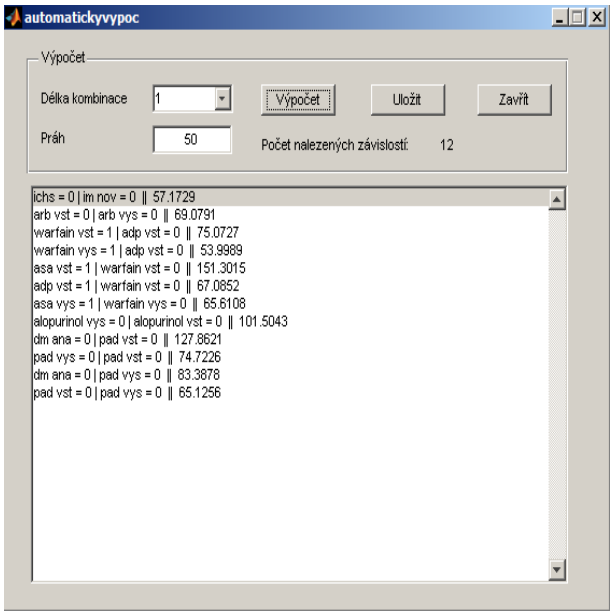
algoritmizaci byla vybrána metoda naivního bayesovského klasifikátoru. Vytvořená aplikace umožňuje na zvoleném datovém souboru hledat závislosti, které odpovídají struktuře asociačního pravidla $Ant \Rightarrow Suc$, kde Ant je předpoklad a Suc závěr pravidla. Aplikace umožňuje tvořit pravidla, která obsahují v části Ant až tři předpoklady, které jsou provázány logickým součinem. Předpokladem i závěrem může být kterýkoli z atributů datového souboru.

Základním problémem je generování všech možných kombinací atributů. Toto generování je realizováno pomocí vnořených cyklů, kde počet takto vnořených cyklů je roven počtu atributů v předpokladu a závěru pravidla. Pro datový soubor o padesáti atributech dostáváme pro předpoklad délky jedna počet kombinací 4900 (musíme si uvědomit, že atributy mohou nabývat dvou hodnot). Pro předpoklad délky dva je počet kombinací roven 235 200, zde jsou započítány všechny možné kombinace hodnot v atributech, které tvoří předpoklad, a platí, že závěrem pravidla se může stát kterýkoli z atributů, které nebyly použity v předpokladu pravidla. Pro předpoklad délky tři je počet kombinací roven 7 369 600, zde jsou započítány všechny možné kombinace hodnot pro atributy tvořící předpoklad a uvažujeme, že závěrem pravidla se může stát kterýkoli ze zbývajících atributů. Celkový počet kombinací délky jedna až tři je pro datový soubor o padesáti atributech roven 7 609 700.

U všech generovaných pravidel je počítána příslušná hodnota naivního bayesovského klasifikátoru, tedy hodnoty aposteriorní pravděpodobnosti pro příslušné kombinace atributů. Pro tento výpočet byly navrženy tři funkce, které realizují generování kombinací a výpočet aposteriorních pravděpodobností. Jsou to funkce: *autonbk1*, *autonbk2* a *autonbk3*. Pro všechny tyto funkce je vstupem datový soubor a hodnota kritéria. Kritériem je poměr mezi hodnotami aposteriorní pravděpodobnosti pro zvolenou kombinaci atributů. Na základě zkušenosti je doporučená hodnota kritéria pro kombinace předpokladů délky jedna rovna 20, pro délku dva rovna 100–500 a pro délku tři rovna 1000–3000. Aplikace na obr. 3.12 zobrazuje všechna nalezená pravidla podle zadaných kritérií, dále vypisuje celkový počet pravidel. Umožňuje také uložení výstupu do tabulky programu Microsoft Excel, kam jsou ještě dodatečně zkopírovány informace o významu hodnot jednotlivých atributů.

3.3.6 Asociační pravidla

Hledání závislostí v datech je možné realizovat také pomocí asociačních pravidel. Asociační pravidla jsou kombinace libovolných atributů v podobě Ant, potom Suc, kdy Ant označuje předpoklad pravidla, který může být tvořen i více atributy a Suc představuje závěr pravidla. Charakteristikou pravidel jsou deskriptivní veličiny pod-



Obr. 3.12: Automatické hledání závislostí

pora, spolehlivost, pokrytí, a další. Nejjednodušším způsobem pro vyhledávání pravidel je testování všech možných kombinací atributů a vyhodnocování výše uvedených deskriptivních veličin. Tento proces je časově náročný, protože počet kombinací dosahuje až stovky tisíc. Pro potřeby hledání závislostí v registru IKK byly vytvořeny dva skripty, které provádí hledání a vyhodnocení pravidel s předpokladem délky jedna nebo dva. Pro vyhodnocení, zda je pravidlo zajímavé se používá veličina spolehlivost, která je definována podle vztahu (2.19), v případě velkého množství nalezených pravidel jsou preferována pravidla s větším zastoupením v datovém souboru. Tomuto zastoupení odpovídá veličina podpora podle vztahu (2.18). Pro tento výpočet byly vytvořeny funkce *maticezamen* a *maticezamen2*, které pro zvolené atributy počítají hodnoty kontingenční tabulky 2.2. Výstupem této funkce je vektor deskriptivních charakteristik daného pravidla. Tato funkce je volána ve vnořených cyklech, které realizují výběr všech kombinací atributů datového souboru. Mezní hodnota spolehlivosti, která odděluje vyhodnocená pravidla, byla nastavena empiricky na 0,95. Tato metoda není zpracována formou aplikace, protože prohledávání kombinací je časově náročný proces a pro většinu datových souborů stačí provést tuto analýzu pouze jednou.

3.3.7 Ověření realizovaných metod

Pro ověření realizovaných metod byl vybrán příklad z publikace [17]. Úkolem je určit klasifikační model nebo soubor pravidel pro správné třídění lebek Tibeťanů nalezených na dvou různých bojištích. Příklad byl zaměřen na klasifikaci na základě

lineární diskriminační funkce (LDA), viz [17]. V tomto případě poslouží tento příklad na dokázání funkčnosti jednotlivých metod a pomocí numerického výpočtu je ověřena správnost návrhu metod.

Zadání úlohy

Úkolem je použitím vhodné metody správně zatřídit doposud neznámou lebku na základě hodnot jejich atributů. Sledované atributy jsou *Ldélka*, *Lšířka*, *Lvýška*, *Ovýška*, *Ošířka* a *Původ*. Atributy *Ldélka*, *Lšířka* a *Lvýška* udávají největší rozměry lebky v milimetrech, atribut *Ovýška* udává výšku horní části obličeje v milimetrech, atribut *Ošířka* udává šířku lebky mezi lícními kostmi v milimetrech a atribut *Původ* udává původ lebky, kdy hodnota 1 odpovídá lebkám z pohřebiště v Sikkimu a hodnota 2 odpovídá nalezištím v okolí Lhasy. Dostupné údaje 20 lebek jsou uvedeny v tab. 3.1. Naším úkolem je klasifikovat lebky, které mají údaje zapsány vektory hodnot $L_1 = (162,5; 139,0; 131,0; 62,0; 126,0)$ a $L_2 = (200; 139,5; 143,5; 82,5; 146,0)$. Původní příklad byl mírně pozměněn, podle originálního zadání se klasifikovala jedna lebka, která byla součástí datového souboru o 32 případech. V našem případě klasifikujeme dvě lebky, které se nevyskytují v datovém souboru o 20 případech. Cílem je ukázat klasifikaci dosud neznámých příkladů.

Transformace dat

Pro všechny realizované metody s výjimkou umělých neuronových sítí je potřeba spojité vstupní atributy převést na atributy binární. Tato transformace je realizována pomocí navržených aplikací. V prvním kroku je pro jednotlivé atributy určena mezní hodnota, která nejlépe rozděluje uvedené příklady do jednotlivých tříd cílového atributu. Cílovým atributem je veličina *Původ*. Výpočet se provádí na základě hodnoty střední entropie pro všechny možné hodnoty daných atributů. Pro jednotlivé atributy byly obdrženy tyto mezní hodnoty: *Ldélka=180,6*, *Lšířka=139,2*, *Lvýška=123,6*, *Ovýška=71,5* a *Ošířka=135,7*. Podle těchto mezních hodnot byla tab. 3.1 převedena na binární tabulku. Cílový atribut *Původ* byl překódován tak, že původní třídě 1 nyní odpovídá hodnota 0 a třídě 2 odpovídá hodnota 1. Transformované údaje jsou uvedeny v tab. 3.2.

Rozhodovací stromy

Větvení rozhodovacího stromu je dáno atributem, který má nejmenší střední entropii vzhledem k cílovému atributu podle vztahu (2.3). Pro výpočet střední entropie musíme nejprve určit entropie jednotlivých tříd zvoleného atributu podle vztahu (2.2). Prvním sloupcem pro výpočet je atribut *Ldélka*, cílovým atributem je veličina

Tab. 3.1: Charakteristiky lebek

Ldélka	**Lšířka**	**Lvýška**	**Ovýška**	**Ošířka**	**Původ**
172,5	132,0	125,5	63,0	121,0	1
167,0	130,0	125,5	69,5	119,5	1
169,5	150,5	133,5	64,5	128,0	1
175,0	138,5	126,0	77,5	135,5	1
177,5	142,5	142,5	71,5	131,0	1
179,5	142,5	127,5	70,5	134,5	1
179,5	138,0	133,5	73,5	132,5	1
173,5	135,5	130,5	70,0	133,5	1
178,5	135,0	136,0	71,0	124,0	1
180,5	139,0	132,0	74,5	134,5	1
183,0	149,0	121,5	76,5	142,0	2
179,5	135,0	128,5	74,0	132,0	2
191,0	140,5	140,5	72,5	131,5	2
181,0	142,0	132,5	79,0	136,5	2
175,0	153,0	130,0	76,5	142,0	2
196,0	142,5	123,5	76,0	134,0	2
185,0	134,5	140,0	81,5	137,0	2
195,5	144,0	138,5	78,5	144,0	2
182,5	131,0	135,0	68,5	136,0	2
174,5	143,5	132,5	74,0	136,5	2

Původ. Pro názornost tohoto výpočtu je sestrojena čtyřpolní tabulka 3.3, která obsahuje četnosti jednotlivých tříd zvoleného atributu v závislosti na hodnotě cílového atributu.

Nyní podle vztahu (2.2) vypočítáme entropii pro obě třídy obě třídy atributu *Ldélka*.

$$H(Ldélka = 0) = -\frac{10}{13}\log_2\frac{10}{13} - \frac{3}{13}\log_2\frac{3}{13} = 0,7793$$

$$H(Ldélka = 1) = -\frac{0}{7}\log_2\frac{0}{7} - \frac{7}{7}\log_2\frac{7}{7} = 0$$

Střední entropii vypočteme podle vztahu (2.3) jako vážený součet entropií pro jednotlivé třídy zvoleného atributu.

$$H(Ldélka) = \frac{13}{20}\cdot 0,7793 + \frac{7}{20}\cdot 0 = \underline{0,5066}$$

Pro ostatní atributy jsou vypočteny tyto entropie *H(Lšířka)=0,8813*, *H(Lvýška)=0,892*, *H(Ovýška)=0,7042* a *H(Ošířka)=0,5066*. Pro dělení první uzel vybíráme veličinu

Tab. 3.2: Transformovaná tabulka lebek

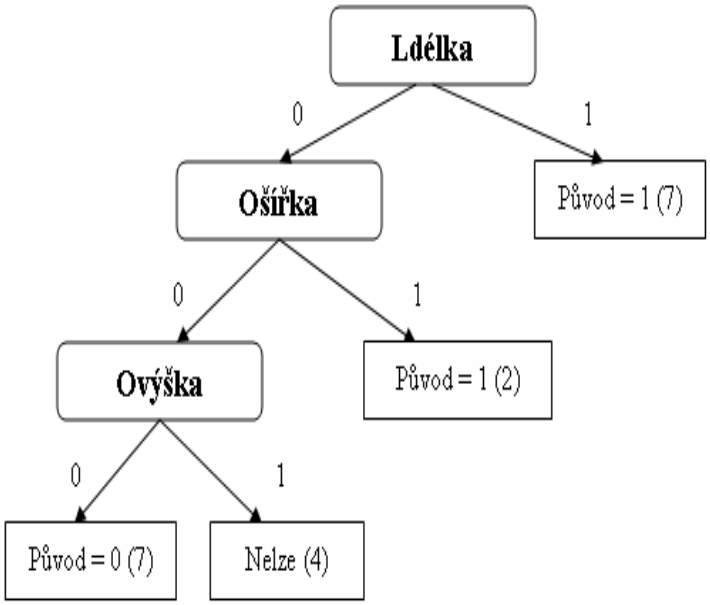
Ldélka	**Lšířka**	**Lvýška**	**Ovýška**	**Ošířka**	**Původ**
0	0	1	0	0	0
0	0	1	0	0	0
0	1	1	0	0	0
0	0	1	1	0	0
0	1	1	0	0	0
0	1	1	0	0	0
0	0	1	1	0	0
0	0	1	0	0	0
0	0	1	0	0	0
0	0	1	1	0	0
1	1	0	1	1	1
0	0	1	1	0	1
1	1	1	0	0	1
1	1	1	1	1	1
0	1	1	1	1	1
1	1	0	1	0	1
1	0	1	1	1	1
1	1	1	1	1	1
1	0	1	0	1	1
0	1	1	1	1	1

Tab. 3.3: Čtyřpolní tabulka pro atribut Ldélka

Ldélka	**Původ=0**	**Původ=1**
0	10	3
1	0	7

s nejmenší střední entropií, v tomto případě je to veličina *H(Ldélka)*. Tato veličina rozdělí datový soubor na dvě podmnožiny, prvky s hodnotou *Ldélka=1* mají veličinu *Původ* rovnou jedna, celkem jich je sedm. Druhá podmnožina s hodnotou *Ldélka=0* obsahuje prvky obou množin cílového atributu, proto je potřeba pro tuto podmnožinu provést další dělení. V tomto případě dostaneme hodnoty středních entropií *H(Lšířka)=0,7079*, *H(Lvýška)=0,7794*, *H(Ovýška)=0,4615* a *H(Ošířka)=0,3719*. Pro dělení je tedy vybrána veličina *Ošířka*, která dále rozdělí podmnožinu na dvě podmnožiny, prvky s hodnotou *Ošířka=1* mají veličinu *Původ* rovnou jedna, celkem jich jsou dva. Pro zbylých jedenáct prvků, které ještě nejsou klasifikovány pokraču-

jeme dále v dělení stromu. V tomto případě má nejmenší střední entropii veličina *Ovýška*. Případy, které mají *Ovýška=0* jsou klasifikovány do třídy *Původ=0*. V datovém souboru zůstaly čtyři prvky, které nemůžeme klasifikovat tímto rozhodovacím stromem. Výsledná podoba takto získaného rozhodovacího stromu je na obr. 3.13. U každé větve, kde probíhá rozhodnutí jsou zobrazeny příslušné hodnoty pro rozhodnutí. V listech stromu jsou v závorce zobrazeny počty prvků v dané podmnožině.



Obr. 3.13: Výsledný rozhodovací strom

Získaný rozhodovací strom můžeme převést na následující rozhodovací pravidla:
IF Ldélka=1 **THEN** Původ=1
IF Ldélka=0 **AND** Ošířka=1 **THEN** Původ=1
IF Ldélka=0 **AND** Ošířka=0 **AND** Ovýška=0 **THEN** Původ=0
Pomocí těchto pravidel je L_1 klasifikována do třídy *Původ=0* a L_2 do třídy *Původ=1*.

Pomocí navrhnuté aplikace pro tvorbu rozhodovacích stromů byly získány stejné výsledky středních entropií pro jednotlivé uzly rozhodovacího stromu. Pomocí této aplikace je možné tento strom úplně vytvořit bez grafického výstupu.

Umělá neuronová síť

Pro řešení tohoto příkladu byla vytvořena dopředná neuronová síť se zpětným šířením chyby, která obsahuje pět neuronů ve vstupní vrstvě, dvě skryté vrstvy se

čtyřmi a dvěma neurony. Výstupní vrstva obsahuje jeden neuron, protože rozhodujeme o jedné konkrétní třídě. Převodní funkce pro jednotlivé neurony je nastavena na hodnotu *logsig*, tzn. že výstupem každého neuronu bude reálné číslo z intervalu $\langle 0; 1\rangle$. Trénovacími daty je tabulka binárních hodnot 3.2, klasické testování pomocí testovacích dat se neprovádí. Učení sítě je přednastaveno na 1000 epoch, ale síť je správně natrénovaná již po 250 epochách učení. Správnost klasifikace na trénovacích datech je 95 % pro kritérium 0,5, tzn. že daná síť špatně zařadila jednu lebku z trénovacích dat. Lebka L_1 byla touto sítí klasifikována do třídy *Původ=0* a lebka L_2 do třídy *Původ=1*. Použitím neuronové sítě je dosaženo stejného výsledku jako použitím rozhodovacího stromu.

Naivní bayesovský klasifikátor

Výpočet aposteriorních pravděpodobností je realizován podle vztahu (2.7). Už v tomto bodě je patrný zásadní rozdíl oproti rozhodovacím stromům. V případě rozhodovacích stromů jsme museli získat určité apriorní informace o celém datovém souboru a tyto informace použít pro určení závěru. V případě naivního bayesovského klasifikátoru aplikujeme metodu přímo na konkrétní dvě lebky L_1 a L_2, i zde počítáme s apriorními informacemi z datového souboru, ale neprovádíme prohledávání celého prostoru hypotéz, jako je tomu u metody rozhodovacích stromů.

Zadané příklady převedeme do podoby binárních vektorů podle stejných kritérií, jaká měla binární tabulka. $L_1 = (0; 0; 1; 0; 0)$ a $L_2 = (1; 1; 1; 1; 1)$. Nejdříve vypočítáme jednotlivé apriorní a podmíněné pravděpodobnosti podle vztahů (2.10) a (2.11).
Pro L_1 dostáváme tyto hodnoty:

$$P(Puvod = 0) = \frac{10 + 1}{20 + 2} = 0,5$$

$$P(Puvod = 1) = \frac{10 + 1}{20 + 2} = 0,5$$

$$P(Ldelka = 0|Puvod = 0) = \frac{10 + 2 \cdot 0,6364}{10 + 2} = 0,9394$$

$$P(Ldelka = 0|Puvod = 1) = \frac{3 + 2 \cdot 0,3636}{10 + 2} = 0,3106$$

Tyto a další vypočtené hodnoty podmíněných pravděpodobností pro L_1 jsou uspořádány v tabulce 3.4.

Poslední řádek tabulky odpovídá aposteriorním pravděpodobnostem vypočteným podle vztahu (2.7) jako součin apriorní pravděpodobnosti a dílčích podmíněných pravděpodobností. Vybíráme třídu s větší aposteriorní pravděpodobností, tedy

Tab. 3.4: Tabulka pravděpodobností

	Původ=0	**Původ=1**
P(Původ)	0,5	0,5
P(Ldélka=0—Původ)	0,9394	0,3106
P(Lšířka=0—Původ)	0,6667	0,3334
P(Lvýška=1—Původ)	0,8560	0,8106
P(Ovýška=0—Původ)	0,6515	0,1818
P(Ošířka=0—Původ)	0,9394	0,3106
NBK	**0,1641**	**0,002372**

Původ=0 jako pravděpodobnější. V tomto případě je hodnota poměru mezi aposteriorními pravděpodobnostmi rovna 69,227. Pro L_2 jsou vypočtené aposteriorní pravděpodobnosti rovny 0,00051 pro třídu *Původ=0* a 0,0951 pro třídu *Původ=1*. Jako pravděpodobnější vybíráme třídu s větší aposteriorní pravděpodobností *Původ=1*.

Pomocí navrhnuté aplikace pro výpočet aposteriorních pravděpodobností byly získány stejné výsledky jako v tab. 3.4. I pomocí naivního bayesovského klasifikátoru jsme obdrželi stejné výsledky pro klasifikaci neznámých lebek L_1 a L_2. Tyto výsledky se shodují s výsledky uvedenými v [17]. Pomocí rozhodovacího stromu zůstaly čtyři příklady z datového souboru neklasifikovány do žádné třídy cílového atributu *Původ*. Tyto čtyři příklady jsou klasifikovány v binární tabulce stejnými hodnotami. Naivní bayesovský klasifikátor zařadí tyto příklady do třídy *Původ=0*, což odpovídá 75 % úspěšnosti klasifikace. Stejného výsledku je dosaženo pomocí umělé neuronové sítě.

Automatické hledání závislostí

Uvedený příklad byl analyzován pomocí navrhnuté aplikace pro automatické hledání závislostí v datech. Pro pravidla s předpokladem délky jedna bylo nalezeno celkem 15 pravidel, závěru v atributu *Původ* vyhovují tři. Mezní hodnota poměru pro vyhodnocení byla nastavena na hodnotu pět. Získaná pravidla:
IF Ldélka=1 **THEN** Původ=1
IF Ošířka=1 **THEN** Původ=1
IF Lvýška=0 **THEN** Původ=1
První dvě pravidla jsou shodná s pravidly získanými pomocí rozhodovacího stromu, třetí pravidlo je nové. Toto pravidlo splňují dva příklady z datového souboru.

Pro pravidla s předpokladem délky dva bylo nalezeno celkem 26 pravidel, potřebnému závěru vyhovují čtyři pravidla. V tomto případě byla mezní hodnota poměru aposteriorních pravděpodobností nastavena na hodnotu 30. Získaná pravidla:
IF Ldélka=1 **AND** Lvýška=0 **THEN** Původ=1
IF Ldélka=1 **AND** Ošířka=1 **THEN** Původ=1

IF Ovýška=1 **AND** Lvýška=0 **THEN** Původ=0
IF Ošířka=1 **AND** Lvýška=0 **THEN** Původ=1
Je patrné, že všechna pravidla jsou rozšířením již dříve získaných pravidel s předpokladem délky jedna. Pro zadanou mezní hodnotu nebylo nalezeno žádné nové pravidlo.

Pro pravidla s předpokladem délky tři bylo nalezeno celkem 138 pravidel. Z tohoto počtu vyhovuje cílovému atributu devět pravidel. U sedmi z nich platí, že jsou rozšířením předcházejících pravidel. Dvě pravidla jsou nová:
IF Lšířka=1 **AND** Ovýška=1 **AND** Lvýška=0 **THEN** Původ=1
IF Lšířka=1 **AND** Ovýška=1 **AND** Lvýška=1 **THEN** Původ=1
Tato dvě pravidla je možné sloučit do jednoho pravidla délky předpokladu dva, které nebylo nalezeno v předchozím kroku.
IF Lšířka=1 **AND** Ovýška=1 **THEN** Původ=1
Toto pravidlo pokrývá sedm příkladů z datového souboru.

Pomocí takto získaných pravidel nemůžeme klasifikovat první příklad L_1 do žádné třídy. Příklad L_2 můžeme podle libovolného pravidla klasifikovat do třídy *Původ=1*.

Asociační pravidla

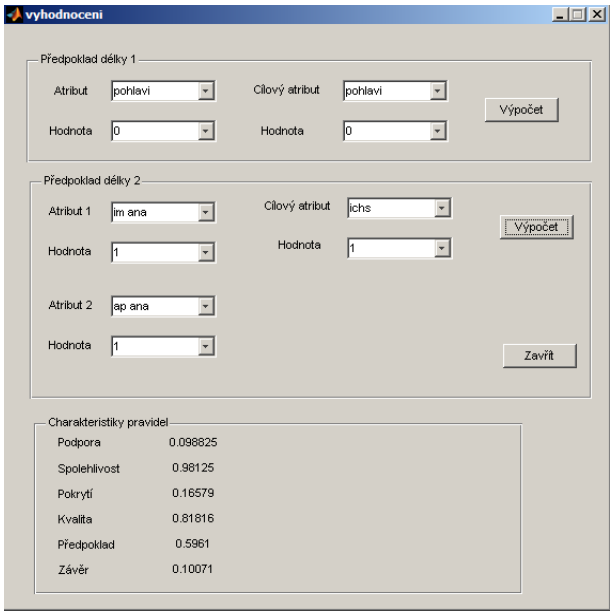
Pomocí uvedeného skriptu je prohledán prostor všech možných kombinací atributů a jsou hodnocena pravidla s předpokladem délky jedna a dva. Kritériem je 95% hodnota spolehlivosti, protože nalezených pravidel není mnoho, není použito doplňující kritérium pomocí podpory nebo kvality pravidla. Pro pravidla s předpokladem délky jedna bylo nalezeno sedm pravidel, z nichž dvě mají cílový atribut *Původ*. Obě pravidla již byla nalezena pomocí jiných metod. Charakteristiky těchto pravidel jsou spolehlivost rovna 100 % a kvalita rovna 94 %. Jsou to tato dvě pravidla:
IF Ldélka=1 **THEN** Původ=1
IF Ošířka=1 **THEN** Původ=1
Pro pravidla s předpokladem délky dva bylo nalezeno celkem 136 pravidel, z nichž 32 má cílový atribut *Původ*. Z těchto pravidel byla vyřazena pravidla, která rozhodují o případech popsaných jednoduššími pravidly. Zůstaly tři pravidla, z nichž první z nich bylo již dříve nalezeno pomocí naivního bayesovského klasifikátoru a dvě další pomocí rozhodovacích stromů. Nalezená pravidla:
IF Lšířka=1 **AND** Ovýška=1 **THEN** Původ=1
IF Ldélka=0 **AND** Ovýška=0 **THEN** Původ=0
IF Ošířka=0 **AND** Ovýška=0 **THEN** Původ=0

Z uvedeného příkladu je patrné, že hledání závislostí v datech je proces náročný

jak časově, tak i s ohledem na vyhodnocení. Obzvláště v případě, že je prováděno hledání naslepo, kdy není preferován konkrétní cílový atribut. Rozhodovací stromy poskytují dobré výsledky pro určité dobře oddělitelné třídy dat, jakými medicínská data už ze své podstaty nejsou. Pro medicínská dat je lepší použít metod naivního bayesovského klasifikátoru nebo asociačních pravidel i za cenu velkého množství nalezených pravidel a nutného zpracování získaných znalostí.

3.4 Hodnocení znalostí

V rámci procesu dobývání znalostí z databází je kladen velký důraz na zhodnocení získaných znalostí. Nezávisle na typu úlohy musí být získané znalosti, popř. modely, podrobeny testování. Pro klasifikační modely se nejčastěji používá hodnocení správnosti klasifikace, popř. některých statistických veličin, jako je např. senzitivita a specificita. Pro získané závislosti formou pravidel je nejčastějším způsobem vyhodnocení určení deskriptivních charakteristik závislostí. Používají se stejné veličiny jako při hodnocení asociačních pravidel. Jsou to podpora, spolehlivost, pokrytí, kvalita a hodnocení relativní četnosti výskytu předpokladu a závěru pravidla v datovém souboru. Pro hodnocení závislostí byla navržena jednoduchá aplikace, která pro libovolné pravidlo s předpokladem délky jedna nebo dva spočítá tyto veličiny. Pro samotný výpočet byla použita již popsaná funkce *maticezamen*. Vzhled aplikace pro hodnocení znalostí je na obr. 3.14.



Obr. 3.14: Aplikace pro hodnocení znalostí

4 ANALÝZA DAT

4.1 Předzpracování dat

Úkolem předzpracování dat je seznámit se s charakterem a strukturou dat a připravit datové soubory pro další práci. Registr IKK Fakultní nemocnice Brno Bohunice (dále jen registr IKK), je dvourozměrná tabulka, která obsahuje záznamy 16 370 pacientů ve 124 atributech. Datový soubor byl upraven podle výše popsané struktury pro jednodušší práci. U atributů, kde je to možné, byl zobrazen histogram rozložení hodnot a primárně byl zkoumán výskyt nulových hodnot. Tyto nulové hodnoty je pro další práci potřeba ošetřit. Většina z atributů, které jsou zapisovány po celou dobu existence registru IKK, má méně než 10 % nulových hodnot. V případech zápisu některých laboratorních hodnot je počet nulových hodnot zhruba 50 %. Tato hodnota je zejména dána faktem, že tyto laboratorní hodnoty nejsou od počátku roku 2008 zapisovány. Další problém je v nesourodosti zapisování diagnózy pacienta, do února 2006 podléhala tato informace internímu kódování registru, od pozdějších dat jsou diagnózy zapisovány podle mezinárodního číselníku diagnóz MKN 10. Kvůli velkému rozsahu číselníku diagnóz není tato informace dále zpracována. Dále jsou v datovém souboru obsaženy atributy, které nemají pro další práci význam, příkladem může být údaj o zapisujícím lékaři.

Nejjednodušší metodou ošetření nulových hodnot je nezahrnout je do datového souboru. Dalšími metodami je nahrazení nulové hodnoty střední hodnotou dané veličiny nebo mediánem. Tento přístup se v medicíně kvůli unikátnosti dat nedoporučuje. Pro další analýzu byly pomocí navržených aplikací vytvořeny datové soubory, které nulové hodnoty neobsahují. Tímto způsobem byly vytvořeny tři konzistentní tabulky. První tabulka obsahuje záznamy 11 379 pacientů v 64 atributech. Tato tabulka obsahuje především anamnestické údaje, údaje o hospitalizaci a užívané léky. Z laboratorních hodnot obsahuje pouze údaje o systolickém a diastolickém krevním tlaku a tepové frekvenci. Druhá tabulka obsahuje záznamy 874 pacientů v 84 atributech. Na rozdíl od prvního souboru obsahuje navíc tato tabulka všechny laboratorní veličiny, které byly zapisovány. Třetí soubor obsahuje údaje 4 766 pacientů v 48 atributech. Atributy do tohoto souboru byly voleny tak, aby apriorní pravděpodobnosti výskytu hodnot byly minimálně 0,1. Tato skutečnost se ukázala být velice významnou při dalším hledání závislostí, protože metoda naivního bayesovského klasifikátoru má tendenci zvýrazňovat atributy s nižšími apriorními pravděpodobnostmi.

Uvedené tři tabulky byly dále pomocí navržené aplikace transformovány na binární tabulky. Pro atributy s diskrétním rozdělením bylo standardně použito kódování *0=ne* a *1=ano*. V případě atributů s diskrétním rozdělením, které má více

možných hodnot, byla použita transformace podle doporučení experta. Příkladem takového atributu je atribut *NYHA*, který obsahuje informace o dušnosti pacienta kódované podle standardu NYHA (New York Heart Association) do pěti tříd. V případě atributů se spojitým rozdělením hodnot bylo použito prahování do dvou skupin podle zvoleného kritéria. Zvoleným kritériem byla nejčastěji limitní mez fyziologických hodnot zdravého pacienta. Tvorba binární tabulky dává velký stupeň volnosti, která se může příznivě i nepříznivě projevit v další práci.

4.2 Ověření známé závislosti

Dalším krokem bylo ověření známé kardiologické závislosti na datových souborech. Pro toto ověření byla vybrána závislost pacientů s prodělaným infarktem myokardu a anginou pectoris na atribut ischemická choroba srdeční. Danou závislost je možné zapsat symbolicky jako pravidlo:
IF $IM\_ana = 1$ **AND** $AP\_ana = 1$ **THEN** $ICHS = 1$

Tato závislost byla ověřena na všech třech datových souborech pomocí metod naivního bayesovského klasifikátoru, rozhodovacích stromů a asociačních pravidel. V rámci explorativní analýzy byla tato závislost také prokázána histogramem. Ověření pomocí naivního bayesovského klasifikátoru je nejjednodušší. Levá strana pravidla tvoří pozorovanou evidenci a pravá strana je cílovým atributem. V prvním datovém souboru je hodnota poměru mezi aposteriorními pravděpodobnostmi 509,2, což poukazuje na silnou závislost mezi předpokladem a závěrem pravidla. V případě rozhodovacích stromů použijeme předpoklad pro vyčlenění datového souboru a sledujeme entropie ostatních atributů na cílový atribut ICHS. Pro danou kombinaci předpokladů se první datový soubor zúží na 914 evidencí, které odpovídají splnění předpokladu. Pro tuto podmnožinu jsou hodnoty střední entropie pro ostatní atributy velmi nízké, což odpovídá tomu, že v dané větvi se nachází většina případů pokrytých závěrem. Použitý algoritmus navrhl pro další větvení (tedy jako další předpoklad pravidla) atribut odpovídající fibrilaci síní. Tato závislost byla také analyzována pomocí aplikace sloužící k vyhodnocení pravidla. Na prvním datovém souboru byly vypočteny tyto hodnoty: spolehlivost 97,8 %, kvalita 81 %. Tento údaj nám říká, že 97,8 % případů, které jsou popsány předpokladem pravidla mají splněn i závěr pravidla.

Hodnoty, které byly získány při ověření známého pravidla, poslouží při další práci jako hodnoty, podle kterých budou nastaveny mezní hodnoty pro vyhodnocení automatického generování závislostí.

4.3 Modelování

Pomocí navržených aplikací bylo realizováno modelování, tedy aplikace dataminingových metod na konkrétní úlohu. Použití metody rozhodovacích stromů je v případě slepého hledání, kdy není předem určený cílový atribut, zdlouhavé. Pro každý datový soubor se předpokládá, že cílovým atributem může být kterýkoli z dostupných atributů a získané rozhodovací stromy jsou podle uvedeného algoritmu často velice komplikované a ani použití metody prořezávání stromů nevede k dobrým výsledkům. Tato metoda se pro hledání závislostí v datech příliš neosvědčila.

Použití metody největší věrohodnosti je omezeno tím, že metoda je navržena pro zkoumání vlivu jedné evidence na cílový atribut. Tento fakt umožňuje získat pouze rámcový přehled o vztazích mezi atributy. Z této metody vychází metoda naivního bayesovského klasifikátoru, která primárně slouží ke klasifikaci případu do skupiny cílového atributu. V tomto případě je možné přidat libovolný počet evidencí (charakteristik případu). Drobnou úpravou je výpočet poměru aposteriorních pravděpodobností pro zadanou evidenci. Pomocí tohoto poměru je možné jednoduše vyhodnotit popsaný příklad jako závislost, za dobré závislosti považujeme závislosti s velkou hodnotou tohoto poměru. Tento fakt jednoduchého výpočtu dává dobrý základ pro automatizaci výpočtu této metody pro automatické vyhodnocení závislostí.

Použití umělých neuronových sítí je vhodné pro klasifikační úlohy. Na všechny datové soubory byla použita dopředná neuronová síť se zpětným šířením chyby, která obsahovala ve skrytých vrstvách 25 a 15 neuronů. Počet neuronů ve vstupní vrstvě byl dán počtem atributů v tabulce a výstupním atributem byl jeden zvolený cílový atribut. I v tomto případě se může cílovým atributem stát kterýkoli z atributů v daném souboru. Pomocí této neuronové sítě byly analyzovány všechny tři datové soubory se zvoleným cílovým atributem ICHS. Data byla rozdělena na testovací a trénovací ve stejném poměru. Úspěšnost klasifikace na trénovacích datech byla v průměru 90 %, na testovacích datech byla v průměru 80 % při zadané mezi úspěšnosti 0,2. Pokud byla mez pro vyhodnocení úspěšnosti klasifikace nastavena na 0,5, úspěšnost stoupla na 95 % na trénovacích datech a na 85 % na testovacích datech. Pro jiné zvolené cílové atributy byla úspěšnost nepatrně menší. Pro klasifikaci jsou to na neznámých datech vynikající výsledky. Bohužel analýza chování neuronové sítě je komplikovaná a pro hledání závislostí není možné umělé neuronové sítě použít.

4.4 Automatické hledání závislostí

Pro automatické hledání závislostí v datech byly navrženy a použity dvě metody. První metodou je hledání závislostí pomocí naivního bayesovského klasifikátoru

a druhou metodou je hledání pomocí konceptu asociačních pravidel. Počty nalezených pravidel pro jednotlivé datové soubory pomocí naivního bayesovského klasifikátoru jsou uspořádány v tab. 4.1. Pro pravidla s předpokladem délky jedna byla mezní hodnota poměru nastavena na hodnotu 50 a pro pravidla s předpokladem délky dva byla mezní hodnota poměru nastavena na 1000.

Tab. 4.1: Počty závislostí nalezených pomocí NBK

	délka 1	**délka 2**
soubor 1	1 226	8 248
soubor 2	2 443	53 019
soubor 3	13	4

Počet nalezených pravidel v prvních dvou souborech je příliš vysoký, protože se negativně projevilo použití poměru jako vyhodnocovacího kritéria. U atributů, jejichž apriorní pravděpodobnosti jsou příliš malé (tzn. daná evidence se v tabulce vyskytuje zřídka), jsou i aposteriorní pravděpodobnosti malé. Tento fakt vede ke zvýraznění málo zastoupených atributů v pravidlech. Ve třetím datovém souboru, který byl navrhnut s ohledem na tento fakt je počet pravidel mnohem menší. V tomto datovém souboru můžeme najít pravidla, která jsou statisticky významná. Bližší analýzou zjistíme, že v nalezených pravidlech je mnohem více pravidel, jejichž závěr přísluší hodnotě 0. Tzn. že taková pravidla vylučují nějaký fakt, příkladem takového pravidla může být pravidlo: *Jestliže pacient nemá diabetes melitus, neužívá inzulin.*

Pomocí metody asociačních pravidel byly analyzovány stejné datové soubory. Kritériem byla veličina spolehlivost s mezní hodnotou 0,95. Počty nalezených pravidel délky předpokladu jedna a dva touto metodou jsou zobrazeny v tab. 4.2.

Tab. 4.2: Počet nalezených asociačních pravidel

	délka 1	**délka 2**
soubor 1	595	65 536
soubor 2	1845	450 866
soubor 3	87	9 661

V tomto případě jsou počty nalezených pravidel vyšší než počty pravidel nalezených metodou naivního bayesovského klasifikátoru. Určitého snížení počtu nalezených pravidel lze dosáhnout zavedením dodatečných kritérií, které posuzují jiné deskriptivní charakteristiky, např. podporu či kvalitu. I v tomto případě převládají pravidla, jejichž závěr má hodnotu cílového atributu 0. Při podrobnější analýze výsledku lze zjistit, že velkou část nalezených pravidel tvoří pravidla, která informují

o užívání léku na vstupu a výstupu. Příkladem může být pravidlo: *Jestliže pacient neužívá warfarin při propouštění z oddělení, neužíval ho ani při příjmu.* Tento fakt je dán samotným registrem IKK, který neumožňuje zohlednění užívání léku nasazených během hospitalizace.

Výstupy získané pomocí obou metod byly upraveny tak, aby počty pravidel byly vhodné pro detailní analýzu. Upřednostňována byla pravidla, která referovala k hodnotě cílového atributu 1. V případě pravidel nalezených použitím naivního bayesovského klasifikátoru bylo snížení dosaženo především volbou atributů s apriorními pravděpodobnostmi v rozsahu $\langle 0,1; 0,9\rangle$. V případě asociačních pravidel bylo snížení dosaženo zvětšením kritéria spolehlivosti na hodnotu 0,98 a použitím dodatečného kritéria podpory, které odfiltrovalo velké množství pravidel se závěrem rovným 0. Výsledky byly poslány expertovi na danou oblast – lékaři, který rozhodne zda získaná pravidla odpovídají již známým závislostem nebo se jedná o nové, potenciálně užitečné, závislosti.

ZÁVĚR

V rámci diplomové práce byl analyzován datový soubor z IKK Fakultní nemocnice Brno Bohunice. Analýza probíhala podle metodiky CRISP-DM, která slouží pro dobývání znalostí z databází. V rámci předzpracování dat byly navrženy postupy pro vizualizaci dat, úpravu tabulky a transformaci dat. Pomocí vizualizace dat byly získány základní informace o jednotlivých veličinách včetně počtu nulových hodnot. Právě počet nulových hodnot musel být zohledněn pro další práci. Výstupem předzpracování dat jsou tři upravené datové soubory, které již neobsahují nulové hodnoty. V dalším kroku byly z těchto souborů pomocí popsaných kritérií vytvořeny binární tabulky, které slouží jako vstup pro dataminingové metody.

V rámci modelování byly pro další práci vybrány a realizovány následující metody, rozhodovací stromy, umělé neuronové sítě, metoda největší věrohodnosti, naivní bayesovský klasifikátor a asociační pravidla. Správnost navržených algoritmů byla ověřena na jednoduchém klasifikačním příkladu a doplněna numerickým výpočtem. Všechny metody byly aplikovány na připravené datové soubory. Jako nejlepší pro klasifikaci se jeví umělé neuronové sítě, konkrétně dopředná síť se zpětným šířením chyby. Bohužel činnost takové neuronové sítě se nedá snadno analyzovat, proto se nehodí pro určení závislostí v datech. Jako nevyhovující se na daných datech ukázala metoda rozhodovacích stromů. Výstup této metody byl v podobě komplikovaného rozhodovacího stromu. Dobrých výsledků bylo dosaženo pomocí bayesovských metod, zejména naivního bayesovského klasifikátoru. Pro všechny metody je společné, že na začátku modelování musí být jeden atribut označen jako cílový. Pro hledání obecných závislostí v datech, ale není žádný z atributů preferován, proto byly navrženy postupy pro automatické hledání závislostí.

Pro realizaci hledání závislostí v datech byly vybrány metody naivního bayesovského klasifikátoru a asociačních pravidel. V obou případech jde o metody, které nejsou výpočetně náročné, ale poskytují dobré výsledky. Největším problémem hledání závislostí je rozsáhlost datových souborů, proto je hledání časově náročné. Bylo provedeno hledání pravidel s předpokladem délky jedna a dva, vyhodnocení kvality pravidla bylo v případě naivního bayesovského klasifikátoru provedeno pomocí poměru výstupních aposteriorních pravděpodobností a v případě asociačních pravidel pomocí deskriptivních veličin podpory, spolehlivosti a kvality. Celkem bylo pomocí uvedených algoritmů nalezeno skoro 500 000 pravidel. Pravidla byla vyhodnocena a redukovaný počet předán lékaři IKK jako výstup této práce.

Pro další práci by bylo vhodné zaměřit se na hledání neobvyklých závislostí na specifičtějším datovém souboru, např. skupině pacientů s konkrétní nemocí. Z uvedeného závěru jsem přesvědčen, že zadání je plně splněno.

LITERATURA

[1] *Association rule learning* [online]. Poslední aktualizace 2008-04-08 [cit. 2008-04-30]. Wikipedia. Dostupné z $< http://eng.wikipedia.org/ >$.

[2] BERKA, Petr. *Dobývání znalostí z databází.* Praha: Academia, 2003. ISBN 80-200-1062-9.

[3] BERMAN, J. Jules. *Confidentality issue for medical data miners.* Artificial Intelligence in Medicine 26(2002). Elsevier Science B.V.,2002.

[4] CIOS, J. Krysztof – MOORE, G. William. *Uniqueness of medical data mining.* Artificial Intelligence in Medicine 26(2002). Elsevier Science B.V.,2002.

[5] *CRISP-DM* [online]. Poslední aktualizace 2008-02-10 [cit. 2008-04-30]. Wikipedia. Dostupné z $< http://cs.wikipedia.org/ >$.

[6] *Data mining* [online]. Poslední aktualizace 2008-05-05 [cit. 2008-05-06]. Wikipedia. Dostupné z $< http://eng.wikipedia.org/ >$.

[7] DUMOUCHEL, William. *Bayesian Data Mining in Large Frequency Tables.* The American Statistician, Aug 1999(53).

[8] *Decision Trees – Tutorial* [online]. Poslední aktualizace 2006-12-28 [cit. 2006-12-28]. Data mining server. Dostupné z $< http://dms.irb.hr/tutorial/tut\_dtrees.php >$.

[9] FAYYAD, U. M. aj.. *Advances in Knowledge Discovery and Data Mining.* AAAI Press/MIT Press, 1996. ISBN 9-780262-560979.

[10] HENDL, Jan. *Přehled statistických metod zpracování dat.* Praha: Portál, 2004. ISBN 80-7178-820-1.

[11] HONZÍK, Petr. *Strojové učení* [elektronické skriptum]. Brno, 2006. Dostupné z $< http://www.feec.vutbr.cz/ >$.

[12] Kol. autorů. *CRISP-DM 1.0, Step-by-step data mining guide* [online]. Citováno 2008-05-01. Dostupné z $< http://www.crisp-dm.org >$.

[13] KONONENKO, Igor. *Machine Learning for Medical Diagnosis: History, State of the Art and Perspective.* [online]. Dostupné z $< http://ai.fri.uni-lj.si/xaigor/xaigor.html >$.

[14] KUSIAK, A., et al.. *Data Mining: Medical and Engineering Case Studies.* Proceedings of the Industrial Engineering Research 2000 Conference, Cleveland. May 21-23, 2000.

[15] LAVRAČ, Nada. *Machine Learning for Data Mining in Medicine.* Plenary invited talk at AIMDM 1999, Aalborg, 20-24. June 1999. Berlin: Springer Verlag.

[16] LAVRAČ, Nada. *Selected Techniques for Data Mining in Medicine.* Plenary invited talk at The Second International Conference on the Practical Application of Knowledge Discovery and Data Mining, London, 1998. Artificial Intelligence in Medicine 16(1999). Elsevier Science B.V., 1999.

[17] MELOUN, M. – MILITKÝ, J – HILL, M.. *Počítačová analýza vícerozměrných dat v příkladech.* Praha: Academia, 2005. ISBN 80-200-1335-0.

[18] NILSSON, J. Nils. *Introduction to Machine Learning.* An early draft of a proposed textbook. Department of Computer Science, Standford University. Standford, 1996.

[19] *Rozhodovací stromy* [online]. Poslední aktualizace 2006-03-03 [cit. 2006-04-25]. Wikipedia. Dostupné z $< http://cs.wikipedia.org/ >$.

[20] *Zákon č. 101/2000 Sb.*. Zákon o ochraně osobních údajů a o změně některých zákonů ze dne 4. dubna 2000.

SEZNAM PŘÍLOH

A POPIS SOUBORŮ NA CD

Obsah CD je umístěn ve dvou adresářích — *programy* a *výsledky*. Adresář *programy* obsahuje všechny realizované funkce, skripty, aplikace a datové soubory. Adresář *výsledky* obsahuje výstupy automatického hledání závislostí. Tento adresář je dále členěna podle použitých metod do dvou podadresářů *asociační pravidla* a *NBK*. Dále je na CD umístěna elektronická verze této diplomové práce, která se shoduje s verzí tištěnou, a manuál pro ovládání navržených aplikací.

A.1 Obsah adresáře programy

Zde je uveden seznam všech souborů v adresáři *Programy*, také je zde uveden krátký popis souborů. Nejsou popsány soubory s příponou *fig*, které obsahují informaci o grafickém vzhledu aplikace.

1. **datové soubory**

   bintabulka1.xls – binární tabulka z prvního výběru dat

   bintabulka2.xls – binární tabulka z druhého výběru dat

   bintabulka3.xls – binární tabulka z třetího výběru dat

   data.xls – kompletní data z IKK

   lebky.xls – tabulka z příkladu klasifikace lebek

   lebkybin.xls – binární tabulka z příkladu klasifikace lebek

   novaspecifikace1.xls – upravená tabulka z prvního výběru dat

   novaspecifikace2.xls – upravená tabulka z druhého výběru dat

   novaspecifikace3.xls – upravená tabulka z třetího výběru dat

   vystup.xls – příklad výstupního souboru z automatického hledání závislostí

   datafile.mat – pracovní soubor obsahující data z aktuálního datového souboru

2. **aplikace**

   automatickyvupoc.m – aplikace pro automatické hledání pravidel pomocí NBK

   bintabulka.m – aplikace pro tvorbu binární tabulky

   histogram.m – aplikace pro zobrazení histogramu

   histogramparam.m – aplikace pro zobrazení histogramu s parametrem

nbk2.m – aplikace pro výpočet naivního bayesovského klasifikátoru

rozdelentropie.m – aplikace pro určení mezního bodu pro binarizaci dat

rozhstrom.m – aplikace pro výpočet rozhodovacích stromů

specifikacetabulky.m – aplikace pro úpravu tabulky

verohodnost.m – aplikace pro výpočet metody největší věrohodnosti

vizualizace.m – nadřazená aplikace pro spouštění všech ostatních aplikací

vybertabulky.m – aplikace pro výběr sloupců tabulky

vyhodnoceni.m – aplikace pro vyhodnocení získaných znalostí

xyzobrazeni.m – aplikace pro xy zobrazení

3. **funkce**

aprst.m – funkce pro výpočet apriorní pravděpodobnosti

autonbk1.m – funkce pro automatické hledání pravidel délky 1 pomocí NBK

autonbk2.m – funkce pro automatické hledání pravidel délky 2 pomocí NBK

autonbk3.m – funkce pro automatické hledání pravidel délky 3 pomocí NBK

binvyber.m – funkce realizující binarizaci dat

entropie.m – funkce pro výpočet entropie

isbinartab.m – funkce pro ověření typu tabulky

maticezamen.m – funkce pro výpočet charakteristik asociačních pravidel délky 1

maticezamen2.m – funkce pro výpočet charakteristik asociačních pravidel délky 2

osetreninulhodnot.m – funkce pro ignorování nulových hodnot

otevritxls.m – funkce pro načtení datového souboru

podmpst.m – funkce pro výpočet podmíněné pravděpodobnosti

vyberdat.m – funkce realizující výběr dat dle zadaného kritéria

vypentrop.m – funkce pro výpočet doporučeného bodu dělení pro binarizaci

znamenko.m – funkce generující znaménko kritéria

4. **skripty**

asocprav1.m – skript pro hledání asociačních pravidel délky 1

asocprav2.m – skript pro hledání asociačních pravidel délky 2

skriptUNS.m – skript pro klasifikaci pomocí dopředné neuronové sítě

A.2 Obsah adresáře výsledky

1. **Adresář *asociační pravidla***

   soubor1delka1.xls – tabulka pravidel pro první soubor délky 1

   soubor1delka2.xls – tabulka pravidel pro první soubor délky 2

   soubor2delka1.xls – tabulka pravidel pro druhý soubor délky 1

   soubor3delka1.xls – tabulka pravidel pro třetí soubor délky 1

   soubor3delka2.xls – tabulka pravidel pro třetí soubor délky 2

2. **Adresář *NBK***

   vystup1.xls – pravidla nalezená pomocí NBK délky 1 a 2 pro první soubor

   vystup1uprava.xls – upravený výstupního soubor prvního souboru

   vystup2.xls – pravidla nalezená pomocí NBK délky 1 a 2 pro druhý soubor

   vystup3.xls – pravidla nalezená pomocí NBK délky 1 a 2 pro třetí soubor

B SEZNAM ATRIBUTŮ

Zde je uveden seznam všech atributů v registru IKK s doplňujícím vysvětlením. Ve sloupci rozdělení jsou uvedeny informace o rozdělení veličiny s následujícím kódováním: S — atribut má spojité rozdělení, D — atribut má diskrétní rozdělení a N – dané rozdělení není možno zařadit. Atributy jsou kódovány následovně: vst — při příjmu, vyst — při propouštění a nov — de novo.

B.1 Údaje o anamnéze a aktuálním zdravotním stavu

Číslo	Zkratka	Rozdělení	Význam
1	Číslo	N	Pořadové číslo záznamu
2	pohlavi	D	Pohlaví pacienta
3	delka hospit	S	Délka hospitalizace pacienta
4	vek	S	Věk pacienta
5	cislo chorobopisu	N	Číslo chorobopisu
6	jina dg 1	D	Diagnóza dle interního kódování IKK
7	jina dg 2	D	Diagnóza dle interního kódování IKK
8	jina dg 3	D	Diagnóza dle interního kódování IKK
9	jina dg 4	D	Diagnóza dle interního kódování IKK
10	jina dg 5	D	Diagnóza dle interního kódování IKK
11	jina dg 6	D	Diagnóza dle interního kódování IKK
12	jina dg 7	D	Diagnóza dle interního kódování IKK
13	jina dg 8	D	Diagnóza dle interního kódování IKK
14	jina dg 9	D	Diagnóza dle interního kódování IKK
15	koureni	D	Informace zda pacient kouří
16	echo hospit	D	Echokardiografické vyšetření
17	ptca ana	D	PTCA
18	ptca hospit	D	PTCA
19	cabg ana	D	Bypassové operace
20	direct hospit	D	Direktní PCA
21	revaskul hospit	D	Revaskularizace
22	pocet tepen	D	Počet ošetřených tepen
23	nyha vst	D	Dušnost dle NYHA
24	nyha vys	D	Dušnost dle NYHA

25	nt pro bnp vst	S	Laboratorní vyšetření Nt-pro-bnp
26	nt pro bnp vys	S	Laboratorní vyšetření Nt-pro-bnp
27	be vst	S	Laboratorní vyšetření big endotelin
28	be vys	S	Laboratorní vyšetření big endotelin
29	ass	D	Akutní srdeční selhání
30	chss ana	D	Chronické srdeční selhání
31	chss nov	D	Chronické srdeční selhání
32	chss eti	D	Původ CHSS
33	ichs	D	Ischemická choroba srdeční
34	dkmp	D	Dilatační kardiomyopatie
35	vchv	D	Vrozená chlopenní vada
36	im ana	D	Infarkt myokardu
37	im nov	D	Infarkt myokardu
38	ap ana	D	Angina pectoris
39	ap nov	D	Angina pectoris
40	dm ana	D	Diabetes melitus
41	dm nov	D	Diabetes melitus
42	ht ana	D	Hypertenze
43	ht nov	D	Hypertenze
44	cmp ana	D	Cévní mozková příhoda
45	cmp nov	D	Cévní mozková příhoda
46	cmp hemor	D	Cévní mozková příhoda – hemoragický šok
47	ichdkk	D	Ischemická choroba dolních končetin
48	fisi ana	D	Fibrilace síní
49	fisi nov	D	Fibrilace síní
50	sr	D	
51	pm ana	D	Kardiostimulátor
52	pm nov	D	Kardiostimulátor
53	crt ana	D	Kardiostimulátor s resynchronizační funkcí
54	crt nov	D	Kardiostimulátor s resynchronizační funkcí
55	icd ana	D	ICD
56	icd nov	D	ICD

B.2 Užívané léky

Následující přehledová tabulka obsahuje informace o skupinách léků, které se do registru zapisují. Užívání všech léků odpovídá diskrétním veličinám. Pro každou skupinu léků jsou zapisovány údaje při příjmu a propouštění pacienta.

acei	arb	bb	caa dhp
caa ndhp	thiazid	furosemid	spironolacton
digitalis	propafenon	amiodarone	asa
adp	warfarin	statin	fibrat
nitrat	alopurinol	pad	inzulin

B.3 Některé laboratorní veličiny

Následující tabulka obsahuje laboratorní veličiny, které jsou zapisovány v registru IKK. Kódování atributů je stejné jako v předchozím případě.

Číslo	**Zkratka**	**Rozdělení**	**Význam**
1	tks vst	S	Systolický krevní tlak
2	tks vyst	S	Systolický krevní tlak
3	tkd vst	S	Diastolický krevní tlak
4	tkd vyst	S	Diastolický krevní tlak
5	tf vst	S	Tepová frekvence
6	tf vyst	S	Tepová frekvence
7	hmotnost vst	S	Hmotnost pacienta
8	hmotnost vyst	S	Hmotnosti pacienta
9	vyska	S	Výška pacienta
10	urea	S	Močovina
11	kreat	S	Kreatinin
12	km	S	Kyseliny močová
13	gly	S	Glykémie
14	chol	S	Cholesterol
15	tg	S	Triacylglyceridy
16	hdl	S	HDL cholesterol
17	ldl	S	LDL cholesterol
18	ery	S	Erytrocyty
19	hgb	S	Hemoglobin
20	ef	S	Ejekční frakce
21	ef zpusob	D	Způsob měření ejekční frakce
22	diastolicka dysfunkce	D	Diastolická dysfunkce